

Johnny.Decimal

How to build your own system

00-09 About this workbook	5
01 Overview	6
02 Before you start	8
03 How to get more help	11
10-19 Define your system's scope	13
11 Overview	14
12 Before you start	14
13 The process	19
14 As you complete this process	22
15 Area complete	22
20-29 Discovery	23
21 Overview	24
22 Before you start	24
23 The process	28
24 As you complete this process	30
25 Area complete	32
30-39 Create your areas & categories	33
31 Overview	34
32 Before you start	34
33 The process	36
34 As you complete this process	38
35 Area complete	42
40-49 Build your system	47
41 Overview	48
42 Before you start	48
43 The process	52
44 As you complete this process	55
45 Area complete	58
50-59 The index/JDex	59
51 Overview	60
52 Before you start	64
53 The process	68
54 Method 1: Individual notes, one per index entry	69
55 Method 2: One note, each line being an index entry	78
56 Method 3: A database	84

57 Area complete	85
90-99 Appendix	87
91 Licence	88
92 Copyright	88
93 Worksheets	88
99 Index	94



00-09

About this workbook

Want more help? 

[Click here to watch this area in the online workshop!](#)

01 Overview

Welcome. You've made a decision to become more organised. The world needs more people like you. Thank you.

01.01 How to use this workbook

This workbook describes a *process* that I have thought about a lot. You've just paid money for this process. It might be tempting to skip ahead and just do the bits that sound fun. But not following the process would be silly.

So read this document properly, slowly, and do what it says in the order that it says to do it. This is 10 years of my experience distilled down to these pages. Trust me.

You will be successful.

01.02 Printing

We tried to make this thing look good. It might be nice to print it out. In colour! You recycle, it's okay. Staple it together like a book: two staples along the long edge.

Make it nice and it will be more fun. Show people that being organised doesn't have to be boring.

The worksheets

If you only print one section, print the worksheets at [93 Worksheets](#). You'll record your progress in these and it's more pleasant to do this with a pencil.

01.03 Read it through before you start

Read the whole thing from start to end so that you know where you'll end up. Don't *do* anything, just skim read through it.

You might as well do that now (after you've printed it).

Worksheet action

Action ID : 01.03

01.04 What was that box?

When you see a box like that, it means that there was an action for you, and there's a space on the 'my actions' worksheet at [93.01](#) to record it.

It's something for you to do, or some information to write down.

01.05 What is Johnny.Decimal?

If you don't know what Johnny.Decimal is, you have a little background reading to do first. This guide assumes knowledge of the system.

Go and read at least the pages in the [11 Core concepts](#) category of the site, and ideally the entire [10-19 Concepts](#) area, then come back here.

<https://johnnydecimal.com>

Worksheet action

Action ID : 01.05

01.06 Help! I'm stuck

The best place to ask for help is the Johnny.Decimal forum.

At the end of every area, there's an 'area complete' action box. These contain links to **forum posts** where you can ask questions.

01.07 Definitions¹

▣ *The* Johnny.Decimal system

Is a collection of methods, tools, and behaviours.

▣ *Your* Johnny.Decimal system

Is your implementation of this system, customised to suit whatever it is that you're organising.

Your system is the end-result of this guide. It's a collection of stuff (files, notes, emails), neatly organised, and tracked in your index.

You can organise anything. It doesn't have to be work-related. It might be your personal life, knitting patterns, or memoirs. It doesn't matter.

What's included will become clear when we define your system's **scope**, which is the first activity.

02 Before you start

02.01 How long will this take?

Obviously this depends on the amount and complexity of the stuff that you want to organise. But I would say, on average, you should set aside between 4 and 8 hours of effort. This effort should span a week or two of duration.

You are about to build a system that you will use for a long time. I've been using my personal system for more than a decade. I used the data centre project system that I talk about below for more than 3 years. It still exists a decade later as an archive.

Taking a week to get it right at the start is time well spent. I know you want to start. I know you want to see those sweet, sweet numbers.

But you *must* take your time. I know this because when I have ignored my own advice, I have always regretted it.

¹ Blog post for further reading: 'Project' is now 'system' <https://jdc.m.a1/22.00.0028/>

02.02 Emergency! I have stuff I need to organise *now*

I get it. Maybe you've started a new job and you're really busy.

You have data – files, notes, email – that you need to store and sort *now*. Me telling you that this is going to take time isn't helpful.

☒ You can still be organised

I find disorganisation and frustration comes from not having *any clue* where a thing is. And I don't mean 'which of your folders it's in'. I mean that you wouldn't even know where to start looking.

There are so many places in our digital life that a thing can be.

We take it for granted that we can usually use our browser history or some sort of 'recent files' list to find what we want. And this often works.

But when it doesn't, or when it's unclear whether the file you've found is actually the latest or the only version, that's where the frustration begins.

☒ Pick one place

So your first step is to pick *one place* that your stuff will be while you're working on your system.

If you're at home it might be a folder in your Documents or iCloud Drive. If you're at work it might be a particular folder on the server or a SharePoint document library.

It doesn't matter where it is as long as you have decided.

Worksheet action

Action ID : 02.02

☒ Within that folder, create temporary subfolders

With the following restrictions:

- No folders-within-folders. i.e. your structure is 'one level deep'.
- Keep it broad at this stage. It's okay if each of these folders contains quite a few items: we'll be re-arranging them when we've created our system. So prefer fewer folders with more things in each of them.
- Don't use numbers. Don't even think about numbers.
- Don't worry about your future system. You're just gathering. There is no expectation that anything you create here will be permanent.

If you're keeping notes – which we'll eventually include in our new Johnny.Decimal system – just make sure that they have descriptive titles. Again, no numbers.

02.03 Help me convince other people

You might be the organised one. But it's likely that if you suggest reorganising a shared file system – and then make other people adhere to this new thing – you're going to get pushback.

It's happened to *me* and I'm Johnny.Decimal!

☒ Lead by example

You just have to take control, and lead by example. Nobody else wants to sort the file system, because they wouldn't know how to. So as soon as you start showing them results, they'll take notice.

Because look at your existing file structure. It's a shambles and everybody hates it. And if it's anything like those that I've seen, people already use numbers at the front of folders. They just do it in an unstructured way.

☒ Numbers beat letters

They do this because the alphabet is a terrible way to organise things. It's ambiguous: was it 'investments', 'budget', or 'savings' that I was looking for? And things move around as items are created. So you never achieve the 'muscle memory' of knowing where a thing is, spatially.

Numbers, even if they're totally unstructured, solve the spatial problem.

Typically I see people start at 01 and then just name each new folder incrementally. But often only the first-level uses numbers and then it devolves in to the usual chaos.

(Numbers still don't prevent people from creating the ambiguous 01 Investments, 02 Budget, and 03 Savings.)

☒ The wow factor

So other people are already on this journey. Help them to do it properly.

In my experience, the reaction to a well-organised system is universally positive. Every single person who has seen one of my systems has the same reaction: "wow!"

02.04 This is a hard problem

Remember that this isn't easy. Humans have failed at this *since we invented computers!* Every file system I've ever seen at every job has been a disaster.

Before we invented computers this was a profession. People filed things *for a living*. You got training. You became an expert.

<https://jdcml.al/video/training/>

So don't be disheartened. Keep at it. Somebody has to, for everyone's sanity.

03 How to get more help

03.01 Our websites

Learn all about Johnny.Decimal on the main site: <https://johnnydecimal.com/>

Access pre-built systems and courses at JDHQ: <https://jdhq.johnnydecimal.com/>

03.02 Johnny.Decimal University

The place to learn all things Johnny.Decimal, including the 5-hour workshop:

<https://jdcm.al/14.02/>

03.03 Life Admin System

A pre-built system that handles the life admin we all need to manage:

<https://jdcm.al/15.03/>

03.04 Small Business System

A pre-built system that handles the paperwork all businesses need to manage:

<https://jdcm.al/15.04/>

03.05 Join our supportive (free!) community

There's a whole community of friendly, helpful people behind Johnny.Decimal. So please ask if you have a question or just come and say hi.

- Forum: <https://jdcm.al/23.01/>
 - Discord: <https://jdcm.al/23.02/>
-

03.06 Stay in touch

Subscribe or follow to get more help and updates.

- Blog: <https://jdcm.al/22.02/>
- Mailing list: <https://jdcm.al/21.02/>
- RSS feed: <https://jdcm.al/21.03/>
- YouTube: <https://www.youtube.com/@johnny.decimal/>
- LinkedIn: <https://www.linkedin.com/company/johnny-decimal/>

10-19

Define your system's scope

Want more help? 

[Click here to watch this area in the online workshop!](#)

11 Overview

This guide's purpose is to help you create a Johnny.Decimal system for 'a collection of stuff', so it is crucial that you understand the boundaries of what you want to organise.

▣ **If we do not carefully define 'the system' we will either:**

- Omit things that should have been included, or
- Include things that we should have omitted.

We design our system according to what it contains: the structure of your areas and categories depend entirely on what it is that you're organising.

Therefore it is important to have the best possible view of everything at the start.

Many Johnny.Decimal systems expand with time, but the more certain you can be at the start the better the result.

Defining what the system will include also gives us mental clarity.

12 Before you start

12.01 How small can a system be?

There is no lower bound to how small a system can be. Although if the thing you're organising is so small that it only has a few items, you probably don't need Johnny.Decimal.

If you find yourself trying to organise a small 'system', ask yourself if it isn't just an 'area' of a larger system.

12.02 How big can a system be?

There *is* an upper bound to how large a system can be. There are only so many Johnny.Decimal numbers. And a system that is too large will lack clarity and be difficult to manage.

That said, you should try to minimise the number of systems in your life. Every additional system is an overhead: both mentally, and in the tracking of its artefacts.

The largest work project that I have managed with a Johnny.Decimal system was a data centre migration for a large Australian government department.

▣ This project was:

- Ten people,
- Working for three years,
- Across twelve physical locations,
- Dealing with three major vendors from whom we procured,
- About 5,000 pieces of hardware,
- With a total budget of \$60m.

Even then I did not 'use' all of this system's Johnny.Decimal areas.

12.03 Painting vs. remodelling vs. rebuilding

I find a house analogy to be helpful in explaining how big I consider 'a system' to be.

▣ Paint the kitchen

Let's start at the bottom. In Johnny.Decimal world, each of your IDs relates to a specific thing: think of it as a job to be done.

So 'paint the kitchen' is just a job. You need to figure out what colour, what type of paint, then you need to measure the kitchen, figure out how much paint you need, buy the paint and the brushes, and paint the kitchen.

That *sounds* like a lot, but there's no reason all of the artefacts related to this job can't be in one folder, or one note. Or both: your Johnny.Decimal ID

relates to *the job*, not specifically to the folder on your file system or the note in your notes app.

You might end up saving half a dozen files. That's fine, because they're all related to the 'paint the kitchen' job. The job itself shouldn't take you longer than a weekend. It's just an ID.

13.15 Paint the kitchen

▣ Remodel the kitchen

So let's remodel the kitchen. We're going to tear it all out: cupboards, appliances, the walls, floor, everything. The kitchen becomes an empty shell and we start again.

Clearly this won't fit inside a single ID. "You need a whole system!", you're shouting. And I really don't think you do. I think you need a category.

Remember what you get in a category: 100 IDs. So does 'redecorating the kitchen' have more than 100 distinct jobs to be done? I doubt it. If those jobs to be done were all contained within one broader folder, would you be able to find them all? I think so.

Would breaking it up any more help you, or get in the way? I'd argue that you'd be creating folders for the sake of it.

So let's give you a category, and you can make your own IDs and get on with the job. You'll be done in a month.

43 Remodel the kitchen

▣ Rebuild the house

Looking at the house, there's no point spending money on the kitchen. The whole place needs to be torn down and rebuilt.

Do you need a full system yet? *Maybe*, but I still don't think so. I think you need an area.

Remember what you get in an area: ten categories, each of which contains 100 IDs. So does 'rebuild the house' have more than ten **categories of things** to be done? And do any of those categories have more than 100 tasks? I doubt it.

I imagine you might need categories for architecture, legal compliance, contractor management, finance, utilities/services, and materials/appliances. Don't take my word for it, I've never built a house – but you get the idea.

So here's our area. Sounds like a 12-month job.

50-59 Rebuild the house

▣ Build a farm

Let's stretch this analogy. So what *would* fill a system, you ask?

I reckon if you buy a vacant plot of land out in the bush and plan to build a farm, this might need its own system.

One area builds you the house, as we've seen. Another might deal with connecting your plot to the local infrastructure: a category each for roads, electricity, and water. One more for setting up a working farm, and we might be done with room to spare.

You can use the remaining areas to run the business for the next decade.

▣ Think big

I tell you this because I want you to **think big** with your system's scope.

I see people post examples of their Johnny.Decimal systems and they've got these really specific areas because they've thought too narrowly about what they're organising. That's a sure-fire way to run out of areas.

12.04 Similar, smaller projects: the 'creative' pattern

I have identified an exception to this preference for larger systems. In some scenarios, you may find yourself needing to manage many similar, smaller projects within a system.

▣ For example

As a hobby, you collect vintage synthesisers. You create sound effects for media producers. You create about one a week. After a few years you may have hundreds of individual customers.

Or you run an advertising agency and you have a team of people working with a dozen clients. Again, after a few years you might deal with hundreds of individual jobs.

These 'jobs' don't fit neatly in to a single ID because you still need to track various artefacts, approvals, outputs, and budgets within the job. You do need that extra layer of organisation.

Using areas or categories doesn't really work because you're limited to ten of each, and it's very possible that one client will provide more than ten jobs.

☒ In this case, your goal should be to create a project *template*

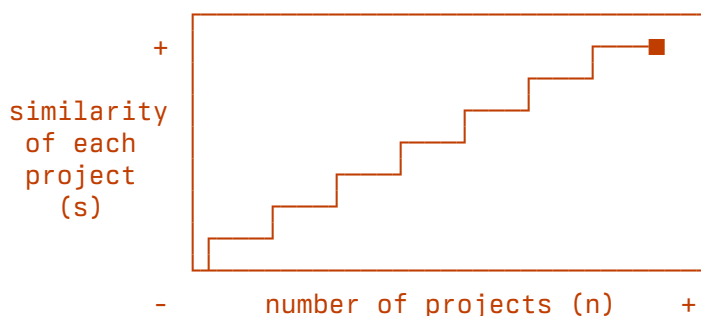
You will use this template for each of the projects that you run.

A good sign of this pattern being right for you is if your projects already have their own 'job code'. This is similar to the point about 'transactional' tasks that I talk about at [34.05](#) below.

I don't go much deeper in to this particular use case in this guide. But if this is you and you're running in to problems, get in touch.

☒ A graph, because it feels scientific

If you were to plot 'the number of projects' across your entire life against 'the similarity of each of those projects to each other', it would look like this.



And this just makes intuitive sense. You can't possibly run that many wildly different projects. Who are you, Kramer?

12.05 Don't think about areas and categories yet

I've talked about areas and categories and how much you can fit in each, because it was necessary to convey the point about how big a system should be.

But at this stage you shouldn't be thinking about your areas and categories. Focus on the overall system until I tell you otherwise.

13 The process

13.01 System scope

Your scope definition is nothing more complicated than a couple of sentences. It tells you, in your words, what's **in** and what's **out**.

Terminology

We call this your system's scope.

Things are either in scope or out of scope.

Your scope statement is so important that you should have it printed, laminated, and stuck to the wall.

If you're running an actual project at work your scope statement should be tattooed on your arm. You should subconsciously murmur it under your breath as you walk down the street. You should wake in the night, sweating from scope statement dreams.

Let's say you're a movie executive for one of the big production companies and you're producing the next superhero movie starring Johnny Depp.

■ *In scope* might be:

- Commissioning the screenplay, including gathering and evaluating submissions from potential writers.
- Building all the sets and props.
- Shooting the movie.
- Editing the movie, including the special effects.

■ *Out of scope* might be:

- Figuring out rights for the franchise. Your studio's legal department will do this for you.

- Anything to do with the physical movie studio. Your studio has a building in Hollywood that's yours for the duration of the movie.
- The cameras that you need to shoot the movie, or the computers to edit it. The studio owns these and they've been allocated to you for the duration.
- Distribution of the movie once the final edit is complete. The studio handles this through their network of movie theatres.

You can see why scope is important. If you *assume* that the studio will supply the cameras, and they don't, that's going to blow out your budget and delay filming.

Defining your scope prevents mistakes.

☒ At home

If you're doing this at home this might sound a bit odd. You just want to organise your day-to-day life administration. And knitting patterns!

Well that's fine, 'organising life administration and knitting patterns' is in scope. But what else?

Maybe you decide to organise more things, in which case add them to your scope statement. Or maybe you have thousands of knitting patterns and you know that this will fill up a system.

Either way is fine, as long as you have defined it before we move on to the next step.

☒ At work

If you're at work this should already be well defined: use the project's scope statement. If you aren't the project manager, go and ask them for it.

(If you're at work and you aren't the project manager, you should be working very closely with them as you create this structure. They're going to love you.)

13.02 Define your system's scope

In the worksheet at [93.02](#), define the scope of your system. This should take between 1–5 days.

These are all of the items that we'll consider throughout the rest of this workbook.

What's in scope is usually pretty obvious. Sentences that describe the thing are good: take inspiration from our movie producer example above.

In the next section we'll do something similar but in a *lot* more detail, so this only needs to be high level.

Regarding out of scope, it is most helpful to note those things that *could have been* in scope, that you have thought about and excluded. Or if you're at work, those things that other people might think are in scope for your project but aren't.

This helps you later. 'I missed something!', you'll think. Then you'll check your scope statement and remember that you didn't miss it: you specifically noted it as being out of scope.

Or someone will say 'why aren't you doing that thing?', and you'll point them to your scope statement and remind them that it's out of scope. (This is why the project manager loves you.)

📌 Your goals

Based on feedback from The Johnny.Decimal Workshop, we've added a box to the worksheet to record goals. This is optional, but others have found it useful. For example:

- Why are you doing this?
- What are you hoping to achieve?
- How do you want to feel when it's done?

Worksheet action

Action ID : 13.02

14 As you complete this process

14.01 Go for walks

I don't want to tell you how to work ... but I'm going to.

This task is well suited to a good walk. You want to let your mind wander, to allow it to consider the broader picture. I find even just walking round the block sparks inspiration – far more so than sitting staring at a screen, anyway.

Your phone will have a way to record a 'voice memo'. Search online to find out how. I use this all the time when I'm walking, and transcribe my notes when I get back home.

14.02 Use a pencil

I know you *can* edit PDFs on your computer, but seriously: print out the worksheet and use a pencil. Seeing the words on the paper in front of you is more real, and it's easy to change.²

15 Area complete



² Pencil nerd side-note: my favourite for this sort of thing is the Mitsubishi 9850.

20-29

Discovery

Want more help? 

[Click here to watch this area in the online workshop!](#)

21 Overview

You've defined your system's scope. It's time to discover the next level of detail.

You will create the structure of your system *entirely* based off the information that you gather here. If it is incomplete, your system will be incomplete. This will be frustrating.

It is vital, therefore, that you take your time.

Within the scope of your system, you are going to write down everything you do for at least **the next week**.

You need your scope worksheet nearby for reference. It is okay to update it if you realise you missed something.

22 Before you start

22.01 Sticky notes vs. digital

I find this task is simpler, more effective, and just more fun if done 'in the real world' with sticky notes. You *can* use online tools – I'll describe how to do this with a mind map – but I strongly recommend using sticky notes.



Get a multi-pack of bright, coloured notes. And sharpen that Mitsubishi 9850, you'll be needing it.

Where do you put these notes? You stick them on your wall!

Of course you can use any sort of paper: 5×7" index cards are another favourite. Sticky notes have the advantage of sticking to a wall. If you use index cards you're going to need a large flat surface or a pack of Blu Tack.

A whiteboard won't work: you need to be able to move the contents around later.

22.02 So what do I write on these sticky notes?

Everything.

It can help to think about this in terms of 'nouns' and 'verbs'. For example, if you worked for NASA, your notes might look like this.

☒ **'Nouns' are the things that you *have*:**

- Minutes from the meeting with the Deep Space Directorate.
- Requirements for the hydrogen gathering probe.
- Test reports from the communications antenna.

☒ **'Verbs' are the things that you *do*:**

- Monitor the progress of the launch pad construction.
- Weekly meeting with the tungsten metal supplier.

☒ **Or *need to do*:**

- Ask the launch team if we can start the countdown at 11, not 10.

☒ **The things you write down might be little anecdotes or gripes:**

- Jim keeps faxing us the launch dates; can he save them in the shared folder?!
- The materials science team keep asking about the colour of the rocket windows ... we've told them five times!

☒ **Or just statements of fact:**

- We decided the end of the probe will be painted with a fake moustache.
- On the morning of launch, it's expected that we'll bring the launch control team cake and coffee.

Depending on your scenario you'll have a mix of these types of notes. Don't worry what that mix looks like: I'm just trying to get you to *think about* all of the different types of things that you might consider.

Do not try to organise your notes at this stage. Just gather as much information as possible.

If in doubt: write a note. You can always discard it later.

22.03 Not just your files

Don't fall in to the trap of focusing on your file system. There are plenty of things that you do that don't generate a Word document.

As we'll see later, Johnny.Decimal's real power comes when you use it to organise everything, and not just your files.

At work, think about the meetings, the phone calls, the conversations in the kitchen. At home, think about places you go, errands you run, businesses or tradespeople you interact with, things you buy or make or plan.

If it happens, stick it on a note.

This isn't to say that you shouldn't reference your existing files. But I find that if you focus on them, they become the focus. Their current structure – if there is any – influences your future structure.

Try not to let this happen. You want to start fresh: if your files were well structured, you probably wouldn't be reading this guide.

22.04 Check your notepad

The notepad that you write things in so that you'll remember to do them and then you flip over to the next page and never read that stuff again?

Go through that. There's tons of useful stuff in there. (You didn't do most of it, but don't let that stress you out now.)

22.05 Pay attention in meetings

Meetings – including informal ‘meetings’ over coffee in the morning – can be a goldmine of information.

Hopefully your meetings are well structured and serve a purpose, but even if they aren’t, they’re full of information. What are people complaining about? Sticky note! What’s running behind schedule? Sticky note! What does Jim from accounts have to complain about this week? Sticky note!

22.06 I’m not running ‘a work project’

If you’re organising your home files, this approach may not feel relevant to you. It’s not like you deal with your house insurance every day.

■ In this case, take a different approach

I still recommend using the sticky notes. They will help you explore your system; having them visually stuck on the wall will help your brain understand the scope of your task.

But switch the approach that you use to write your sticky notes. Rather than writing down the things that you do each day, go through your existing electronic files, paper documents, and written notes.

Make sure you gather all of your things from all of the places that they can be. Don’t forget any archives you have, the files on both your physical and computer’s Desktop, your documents folders, and any of your cloud storage drives.

■ Now is the time to collect and consolidate

You do not need to feel overwhelmed. Sticky notes are your happy place.

Your wall should end up covered in sticky notes that say things like photography, review insurance quotes, study, make garden plan, concert ticket stubs, update meal plan, pet care, taxes, triathlon training, finish toolshed inventory, gaming, holidays!, type out Grandma’s recipes, car stuff ... or whatever it is that you do with your life.

22.07 Consolidate your files (if you want)

Similar to collecting things in to one big folder (as mentioned at [02.02](#) above), you may find it helpful to do the same. Just drag and drop your folders as they are, but drag them all in to one central location. (This is optional.)

22.08 I just started a big new thing, what do I do?

If you're starting from scratch – let's say you just started university or a new job – you might not know what you'll need to organise yet.

This is fine, because the assumption is that you don't already have a bunch of stuff that needs to be organised. Everything's new: clean slate. Nice.

In this case I recommend following this guide. But rather than it being one large process that takes many weeks, **make it lots of small processes that you do more frequently.**

You might set aside a few hours every Monday morning to think about what happened last week. Use all of the techniques in this guide but do them more quickly, more often.

In this case the principle of 'know where everything is', as noted above, is particularly important. Don't let things get out of hand as soon as you've begun! If you see chaos start, rein it in fast.

23 The process

23.01 Spend a week writing sticky notes

This process is lots of tiny processes throughout the week, and they all go like this:

1. When you think of something, write it on a note and stick it on the wall.

And that's it.

23.02 Using a mind map

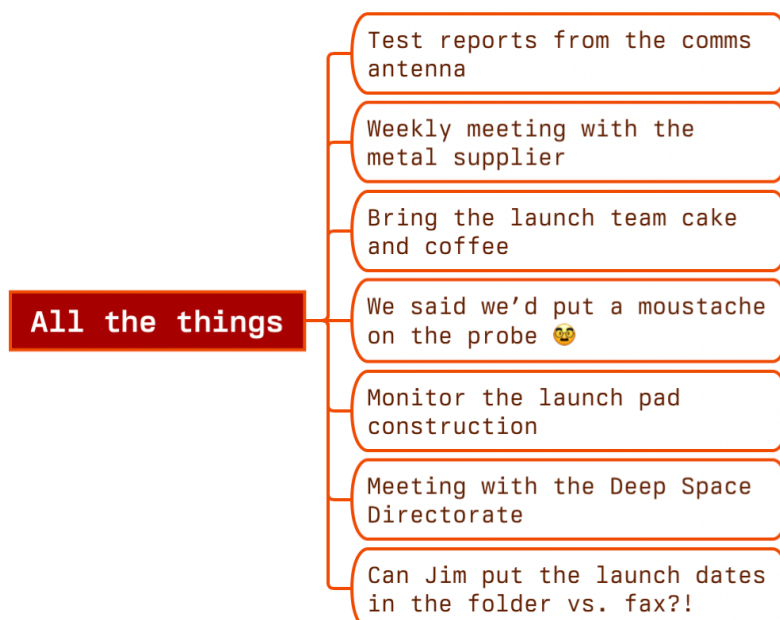
If you prefer to use software rather than sticky notes, a mind map works great. This is also more private if you'd rather not have the contents of your brain stuck to the wall.

I love and recommend MindNode³ for Apple's platforms. It's one of my favourite pieces of software for any purpose, and well worth the annual subscription.

I don't have a recommendation for non-Apple platforms, sorry. Any free app or online tool will work: some are just nicer than others. You don't need any advanced features.

So instead of writing sticky notes, you add new nodes to your mind map.

Add them all as first-level nodes attached to the root. Your mind map should end up looking like this.



As with our physical notes, you must resist the temptation to create structure. Just get everything out of your head for now.

³ <https://mindnode.com>

24 As you complete this process

24.01 Refer to your scope statement

You wrote it for a reason. You should only be writing notes for things that are **in scope** for this system.

If you find yourself wanting to note something that is out of scope, you have three options:

1. **Modify** your scope statement so that it includes the new thing.

This is totally fine, as long as you give it due consideration. And make sure you update the document! Don't just think you will. Otherwise this leads to 'scope creep'.

2. If you're at work, figure out who *does* manage this item, and make sure that it's included there. The project might not exist yet, in which case the item needs to remain in some sort of 'inbox'.

Similarly, if you're at home and you think the item is important but not relevant to this system, make a note for the future.

3. **Drop it!**

You can't organise everything in the world. This thing might not be your responsibility. Don't try to take on too much: it'll overwhelm you and your new system will suffer.

24.02 How many notes do you expect me to write?

Obviously this depends on how much stuff you're organising, but you should be looking at 30 as an absolute minimum. If you do this for a week – say that's five work days – that's only six notes a day. Less than one an hour.

Do you really only do one thing every hour? You're always complaining that you're too busy.

More is better. Sticky notes are cheap. Don't worry if you feel like you're writing down the same thing more than once – it'll all work out in the next step.

24.03 You're really expecting me to do this for a *week*?

As always, 'it depends'.

If this is a work project, I'd say for sure spend a week. A week gives you the full cycle of meetings and reports so you should be confident that you've captured everything.

If you're doing this at home and you already know what needs to be organised, it might not take you a week. But still, give it the time it deserves.

If you use this system for 5 years, 1 week is 0.4% of that time. Isn't it worth the investment to be sure you've got it right? What's the rush?

24.04 Can I...

...organise my notes as I go? Can I start to use colours to indicate things of a similar type? Can I start to group them by moving them around?

No.

24.05 Stare at your sticky notes sometimes

This is why paper-stuck-to-wall is such a good method.

During the day, take time to just stare at your notes. You wanted to spend less time looking at a screen anyway.

Just read them all again. Then go and make a cup of tea. You'll think of something else! Write it on the wall.

Let your brain absorb all of this. Let it spin around in there.

24.06 Use your experience

If you're doing this at work, you already know what a bunch of these notes are going to be. You don't have to wait for them to happen.

If you're a project manager, you know you manage stakeholders and risk.

Alternatively, if you've been running your household for many years, you'll know the recurring themes of your life administration.

24.07 Engage the team

If you're doing this at work, please do it with the rest of the team. They will all need to use this system: involve them early and they'll be much more engaged.

It's probably worth scheduling a session dedicated to this. Do it later in the week so that you can show them what you've already got.

Read over your existing notes and get people to shout out anything else that comes to mind. Perhaps review the project requirements or schedule? Anything that can trigger ideas is good.

Of course, you will find some that are more engaged than others. That's okay: some people are naturally organised, and some aren't. Don't force it.

Worksheet action

Action ID : 24.07

25 Area complete

25 Area complete

Forum help

<https://jdc.m.a1/d41.25.92/>

30-39

Create your areas & categories

Want more help? 

[Click here to watch this area in the online workshop!](#)

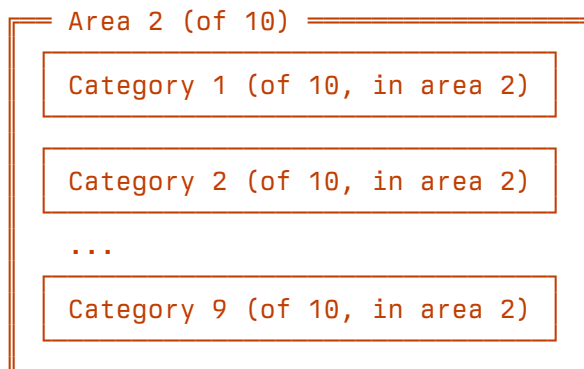
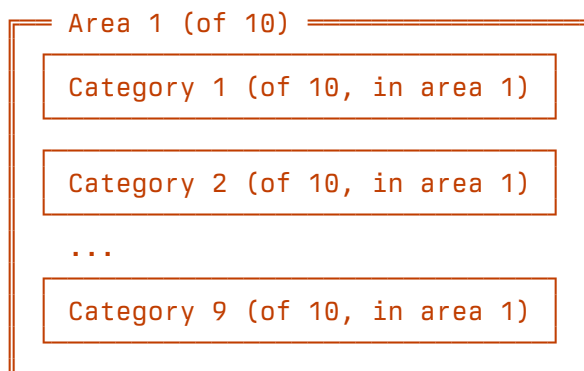
31 Overview

You can guess what's happening next. We're going to take those sticky notes and organise them in to areas and categories.

32 Before you start

32.01 The Johnny.Decimal structure and its constraints

Let's remind ourselves of the kind of structure that we're aiming for.



At the first level of organisation we have *at most ten areas*. And we should strive to use fewer.

This should tell you that these areas are necessarily broad, and indeed this is the point of Johnny.Decimal.

Your goal is to design a system that a stranger could use. Let's set this imaginary person a challenge: 'find my health insurance policy'.

In a system with 50 folders at the first level we are immediately asking them to choose from 50 things. This is already overwhelming. There will *inevitably* be ambiguity. They will fail.

In a system with at most ten areas this choice is obviously easier. If you're looking for the health insurance policy and your three choices are:

- Life administration
- Home business
- Tennis club

Then your choice is clear.

As you design your system, constantly have this in the back of your mind.

Categories are a repeat of this pattern within the context of an area. The same rules apply: make them unambiguous, and strive for fewer.

You're already 'in' an area when you're browsing categories, so you've already set some context.

32.02 What's an 'area'?

Think of these areas as 'areas of your life', or 'areas of expertise'. That's how broad they should be.

I've seen people design systems with very narrow areas. Remember, you only get ten. Narrowly-defined areas mean you'll use them up. You should leave room for expansion.

32.03 What about 'category'?

I might as well quote from my own website here:

*They're collections of similar stuff: travel documents; marketing material for your small business; facilities maintenance for the tennis club ... Any type of 'work' you do can become a category.*⁴

⁴ <https://jdc.m.a1/11.02/>

In the context of your sticky notes, each one is a *thing* that needs to fit in to a category. It's unlikely that you've written any categories on your notes: you've written the items, tasks, and thoughts that need to be *categorised*.

So when you perform this exercise, for each note think about what **type of work** it relates to. Three notes (an item, a task, and a thought) might all fit in to the same category.

Later in this process these individual notes become the IDs that we create in your new categories.

33 The process

33.01 Group your sticky notes

This should feel intuitive: you're going to take all of your notes and group them in to areas or categories.

33.02 Do I do my areas or my categories first?

There are two ways to create your system.

You could group everything in to categories, then group those things in to areas.

Or group everything in to areas, then split those areas in to categories.

My instinct tells me that most people would tend to group by category first. That feels like the way most human brains work: we see the 'local' patterns more naturally than the higher-level patterns.

If you have the luxury of time – and, really, this should only take an hour or two – I *strongly* encourage you to try both methods.

33.03 Home > Insurance, or Insurance > Home?

Doing it both ways might lead to different results; as in, totally different areas and categories. Remember, there isn't a 'right' answer here. This is more art than science.

Take something like your home insurance. Do you classify that as 'Home > Insurance' – that is, 'Insurance' as a category under the area of 'Home' – or 'Insurance > Home'?

It depends! It depends on your brain; it depends on the rest of your system; it depends on whether you're an insurance broker or a real estate agent. And that's cool. Either will work. You just have to pick one and get on with it.

33.04 Name your areas and categories

When you're happy with your groupings, give each of your new areas and categories a name.

Some are going to be really obvious, but others can take a bit of work. Remember that words are free. Humans tend to want to give things shorter names thinking that it's somehow neater, but you should name your areas and categories so that they are as unambiguous as possible.

▣ Involve the team

Words are tricky, especially if you're sharing them with others. People express things in different ways. Everyone's brain is different.

If you're doing this with a team, the people who use an area most should have the biggest say in how it and its categories are named.

We'll see later (at [54.10](#)) that your index can help you with any remaining ambiguity.

Worksheet action

Action ID : 33.04

34 As you complete this process

34.01 How many categories should I have?

There's obviously no 'correct' answer to this, and it's heavily dependent on the type of system you're designing.

You should prefer broader categories. You can always split them later. So if you have two categories and you think that they might work as one, try merging them.

In a typical work project, I've consistently found myself using 3-6 categories in each of 3-6 areas. So that'd be a lower bound of 9 total categories and an upper bound of 36. I haven't been keeping count but my gut tells me that I usually land somewhere around the 25 mark.

I don't think I've ever used more than half of the possible categories in a system. I'm not saying that you can't. But if you find yourself in a 'category explosion' situation, question whether they shouldn't be merged.

▣ Fewer categories reduces decisions later

Every category is a decision that somebody has to make: is it in this one, or that one? The fewer categories, the fewer possible decisions.

Or think of it like this: ideally, you will be familiar with each of your categories. You shouldn't ever look at one of them and think, 'what was *that* about?'

It's easy enough to be familiar with 25 *types of thing* on a work project, or in your life. But once we get up to 50 that gets more difficult, and 100 is near impossible. There's only so much the human brain can handle.

34.02 Divide and conquer

If you do find yourself with 100 categories - it's possible, I'm not saying don't do it - your system should be structured such that some people deeply understand some of them, and other people understand others. This level of understanding should congregate around your areas.

If you're the project manager, you might be very familiar with the categories in the 'project management' area. Meanwhile your technical lead won't have much idea what goes on in there, but will be comfortable in the 'technical' area, about which you have no clue.

In your home life, you might have categories that you rarely touch. You might have archives of material from your childhood. This is okay.

☒ **Remember, our goal is to reduce mental burden**

Just bear in mind that the more *active* categories that you try to manage, the more mental overhead you will inevitably endure.

The goal of Johnny.Decimal is to reduce this mental burden. It is *not* to use up all of the available numbers.

34.03 How many things should each category contain?

As far as how many items should be in each of your categories, again there's no right answer.

But if any of your categories contain a *single* sticky note, perhaps question their validity. Can you think of the next thing that will go in there, even if it doesn't exist yet? Cool, the category is probably valid. If you can't, consider merging it.

34.04 When you're in a category, you're organised enough

In the Johnny.Decimal system, your categories are the unit of organisation.

Above categories we have areas: areas serve to organise categories. But the main thing that your brain is hooking in to, the thing that you will remember, is a category.

Below categories we have IDs. IDs are just items. By definition in Johnny.Decimal we do not 'organise' our IDs: they start at .01 and increment.

This is because, by the time we get to a category, **we are organised enough**. We do not need to go any further.

Looking through IDs within a category should not be difficult. If it is, your category is badly designed.

☒ Sort is our friend

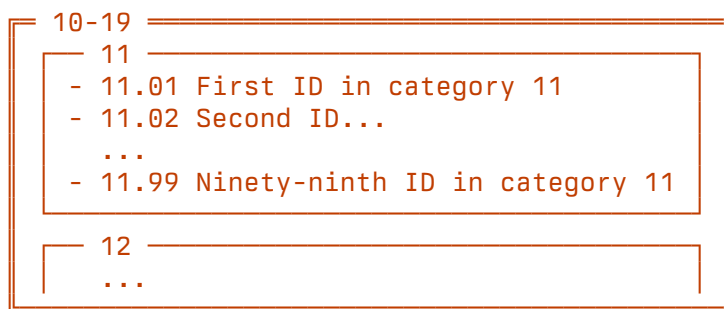
Remember, creating IDs by simple incrementation means that they naturally sort by creation date. Things that were created together, stick together. Things that were created most recently have higher numbers.

It is usually the case that the items you need to work on today are those that were created most recently. Incremental IDs make this easy.

Older IDs just ‘hang around’ together in case we need to refer to them later. But we can often ignore them in our day-to-day work.

34.05 Is one hundred enough things?

When creating categories remember that each can eventually hold 100 items.⁵



For some types of work, you might think that 100 items is not enough. This is possible if this type of work is **transactional**.

If we split all work in to two types, we have:

1. *Transactional*: work that repeats; work that has a pattern.

Think about invoices, incidents, change requests.

These things are easy to spot because they typically already have a unique ID – there is already a system being used to track them. **Use that ID.**

2. *Non-transactional*: everything else.

If you find yourself with anywhere near 100 non-transactional items in a category, the category is by definition *too* broad.

⁵ You will notice that 01-99 is 99 things, not 100. To understand why we don't start at 00, see <https://jdc.m.al/12.03/>

34.06 Look for ways to create standards

As you're designing your categories, look for patterns.

Most professions include tasks that you perform repeatedly. That's what makes it a profession. Find those tasks, categorise them, and use those categories across every work project.

We've already noted that project managers always deal with stakeholders and risk. So on every new project, I use *the same category numbers* for these things (see [44.05](#) below). This won't help you the first time, but by the second and third it becomes a superpower.

34.07 Dealing with conflict

If you're doing this with others at work or home, there will inevitably be some ambiguity in the final result. Life doesn't just fit in to buckets and we all have different views on how to organise a thing.

We see this online: people find one thing wrong with an idea and consign the whole thing to the trash.

Don't let the hand-wavey doomseekers ruin it all. Just because you can't achieve perfect categories doesn't mean that *pretty good* categories aren't better than flat-out chaos.

We're humans. We're smart. We remember things with our brains. It'll be okay.

34.08 Filename length limitations

At work you might find that Microsoft's limit on how long a file path can be will ruin your nice descriptive names.

There's not much you can do about this. I still recommend designing your system with the 'full' names, because you'll store these in your index. You can then adapt your file system as required.

Tip

Replacing a space in a folder name with an underscore saves you *two* characters if your file server is on **SharePoint**.

Now we group them. In this highly contrived example, it feels more natural to group by colour. Doing this forms our areas.

Create your areas

Red things



Green things



Blue things



We take each of these areas and group the items again, giving us our categories.



Notice that we've ended up with duplicate categories named 'animals' and 'symbols'. You could make these more specific and call the category 'blue animals' (or even just 'fish' in this case – but then what happens when a blue iguana comes along?), but you don't need to.

If you challenge someone to find the pair of jeans, there's no ambiguity. Step through the decision process: first, they're presented with three areas. Are jeans usually red, green, or blue? Next, they're presented with four categories: are jeans emoji faces, symbols, animals, or clothing?

35.02 Emoji by category then area

To see how it worked out, I tried grouping these same emoji by category first, and then area. (I constructed this example specifically to work the way we've already seen - I did warn you that it was contrived.)

Here's what I came up with. It *works*, but I definitely prefer the first version. I'll write this out in the same way that I write out my areas and categories in real life: in my day job I'm not normally drawing boxes around emoji.

Living things

Animals

- Frog
- Snake
- Caterpillar
- Tortoise
- Whale (spouting)
- Dolphin
- Whale

People and emoji faces

- Jemima
- Simone
- Booker
- Nicole
- 'Really cold' emoji
- 'Fearful' emoji
- 'Sad' emoji
- Munch's 'The Scream' emoji

Plants

- Shamrock
- Pine tree
- Palm tree

Inanimate things

Things that you wear

- Thread
- Business shirt
- Lipstick
- Casual shirt
- Trousers

Household objects

- Telephone
- Drawing pin

Food and drink

- Glass of red wine

Kiwi fruit
Apple
Cucumber
Sports equipment
Ping-pong racket
Boxing glove
Golf flag

Conceptual things

Symbols and signs
Wheelchair going real fast
'Prohibited' in Japanese
Passport control
SOS
Blood type A
Left luggage
Customs
Blood type B

Let's test this: find the frog. Is it living, inanimate, or conceptual? Easy.
Now is it an animal, person, or plant? Also easy. And there's the frog.

This scheme *works*, I just prefer the first. You might prefer this one, or another variant. There's no right answer.

In the next area we'll use this same list to assign our numbers.



40-49

Build your system

Want more help? 

[Click here to watch this area in the online workshop!](#)

41 Overview

You've done the conceptual work. Now it's time to build your system and start organising stuff.

You'll need the areas and categories that you have recorded on worksheet [93.03](#), along with the sticky notes that you assigned to them.

42 Before you start

42.01 Johnny.Decimal: a refresher

I asked you to read the [11 Core concepts](#) category of the website, and I know you did. So you're familiar with the Johnny.Decimal system.

But it is **really important** that you understand how the numbers work. So even if you think you know this, please read it carefully.

At the heart of Johnny.Decimal is the category. Like I said before, this is the thing that your brain will hook in to.

The system is designed to allow you 100 categories, so they range in number from 00 through 99. (We always use leading zeros.)

To better organise our categories, we group them in to areas. The name of the area reflects the categories that it contains.

So area [00-09](#) contains the categories from 00 through 09.⁷ Area [90-99](#) contains categories 90 through 99.

⁷ As mentioned before, we tend to reserve [00-09](#) for the system: see <https://jdcn.al/12.03/>

This gives us a hierarchy that looks like:

```
00-09 The zeros
  00 Category zero zero
  01 Category zero one
  ...
  09 Category zero nine

10-19 The teens
  10 Category ten
  ...
  19 Category nineteen

...

90-99 The nineties
  90 Category ninety
  ...
  99 Category ninety-nine
```

This is exactly how it looks in your file system. Remember that one of the reasons we use numbers is that they sort predictably. So they don't move around even if you rename the thing that they relate to.

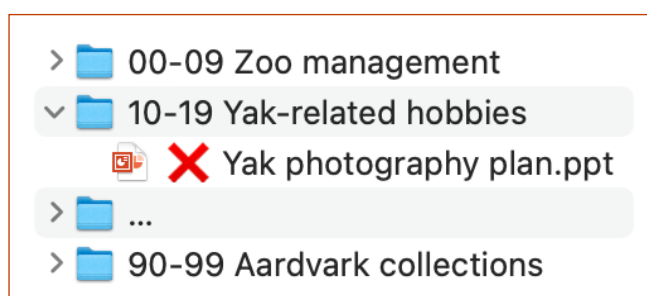
And now the alphabet can't interfere with our sort order.



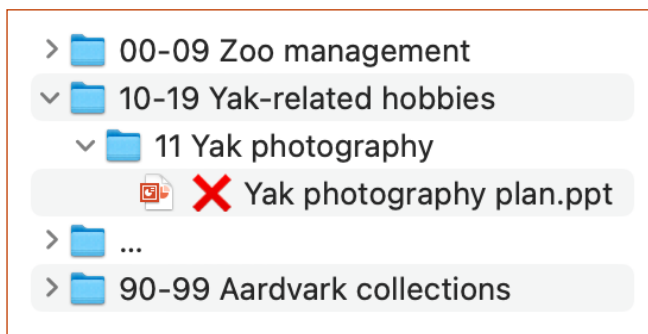
42.02 Areas and categories are containers

One of the important limitations of the system is that we don't store anything in area or category folders. This enforces structure and prevents chaos.

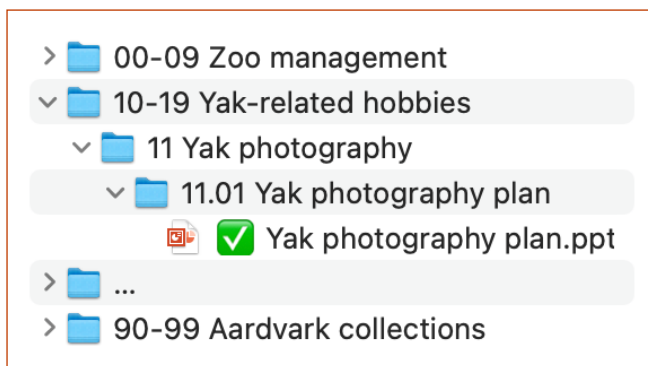
It's absolutely forbidden, for example, to do this.



Or this.



This is what we want.



We say that your Johnny.Decimal structure is ‘only two levels deep’, but it’s more than that: everything you save is *exactly* two levels deep.

You don’t save anything directly under an area. You don’t save anything directly under a category. This makes your system predictable.

In order to save anything – a file, a note, an email – we need to create an ID within a category. And this is where the full Johnny.Decimal ID comes in.

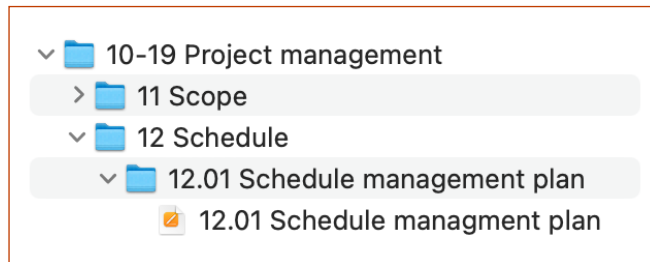
Let’s create an ID in category 12. It’s the first ID we’ve created, so it’s number 01.

12.01

Everything needs a title, of course.

12.01 Schedule management plan

And there we go. Create a folder and save your schedule management plan in it.



42.03 IDs are where we store our stuff

But what exactly is an ID? How much should they contain? When you sit down to do something, your thought process should be:

1. Which category is the thing in?

You will find yourself remembering those you use frequently.

2. Which ID within this category is the thing I need to work on?

It will usually be one of the higher numbers because recent things have higher numbers.

Now you're 'in' an ID and that's where your stuff is. It's the documents from your travel agent for your next trip, the notes and spreadsheets for the meeting with your accountant, or the design files for the tennis club's competition flyer.

The sticky notes that you have written – the things you have, or do, or track, or whatever – will become your IDs.

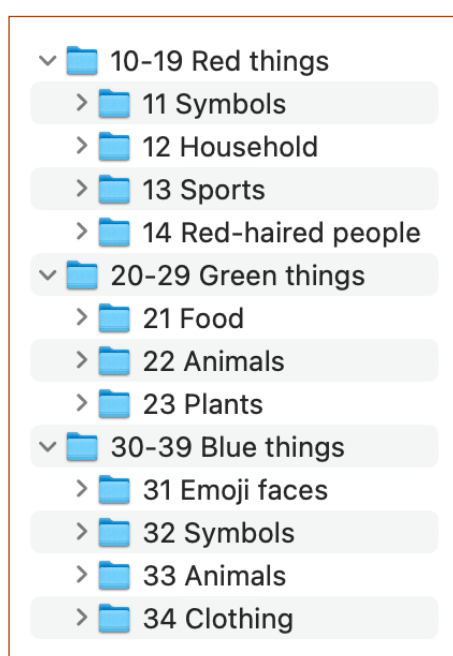
43 The process

43.01 Assign numbers to your areas and categories

Ignoring your IDs – the individual sticky notes – for now, assign numbers to the areas and categories that you have made.

There are two ways to do this.

📁 Create folders in your file system



This is the simplest and most intuitive method.

Create folders in your file system to create your area and category structure.

At this stage I'd like you to do that and then stop. So you end up with something that looks like this.

📄 Use a text file

I'll note this here for completeness, but as I spend the next workbook section talking about 'the index', I'll keep it brief.

If you'd prefer to create your structure in a text file – just like the structure shown at [42.01](#) above – do that instead.

I find the text file method easier to change. Our system isn't locked in yet.

43.02 Time to think again

You've just created the structure that you'll live with for the life of this system.

At this stage I encourage you to stop and think about the system you've created for as long as you can.

At the very least give it a day or two. As before, you should stare at it sometimes and go for a walk. Let your brain work it over. Think about what you're organising and make sure that things 'fit'.

Because it's *possible* to change later, but we'd rather not.

Let's look at how our emoji example from [35.01](#) above would look.

10-19 Red things

- 11 Symbols
- 12 Household
- 13 Sports
- 14 Red-haired people

20-29 Green things

- 21 Food
- 22 Animals
- 23 Plants

30-39 Blue things

- 31 Emoji
- 32 Symbols
- 33 Animals
- 34 Clothing

43.03 When you're *sure*, create your IDs

When you're happy with your areas and categories, create your IDs. You need an ID for each unique thing that you identified on a sticky note.

You might find yourself consolidating some of your notes. Perhaps you wrote two notes about the same thing: one was a reminder that the thing existed, and another was an action to update the thing.

In the next example, we've created a single ID for the two 'blood type' emoji at [11.03](#), and consolidated our fish and whales in category [33](#).

10-19 Red things

11 Symbols & signs

- 11.01 'Prohibited' in Japanese
- 11.02 SOS
- 11.03 Blood types

12 Household things

- 12.01 Telephone
- 12.02 Drawing pin
- 12.03 Lipstick
- 12.04 Glass of red wine

13 Sports equipment

- 13.01 Boxing glove
- 13.02 Table tennis bat
- 13.03 Golf flag

14 Red-haired people

- 14.01 Jemima
- 14.02 Simone
- 14.03 Booker
- 14.04 Nicole

20-29 Green things

21 Food

- 21.01 Apple
- 21.02 Cucumber
- 21.03 Kiwi fruit

22 Animals

- 22.01 Snake
- 22.02 Frog
- 22.03 Caterpillar
- 22.04 Turtle

23 Plants

- 23.01 Shamrock leaf
- 23.02 Pine tree
- 23.03 Palm tree

30-39 Blue things

31 Emoji

- 31.01 Really cold
- 31.02 Fearful
- 31.03 Sad
- 31.04 The Scream?

32 Symbols

- 32.01 Left luggage
- 32.02 Customs
- 32.03 Passport control
- 32.04 Wheelchair going real fast

33 Animals

- 33.01 Dolphin
- 33.02 Fishes
- 33.03 Whales

34 Clothing

- 34.01 Business shirt
- 34.02 Casual shirt
- 34.03 Thread
- 34.04 Trousers

Worksheet action

Action ID : 43.03

43.04 Move your files

Assuming you've created folders in your file system, you can now move your existing files into these new folders.

Now's a good time to tidy them up a bit. Our Support Centre has some sensible file-naming strategies.⁸

Worksheet action

Action ID : 43.04

44 As you complete this process

44.01 How many things do I save in each folder?

One or more.

How many more? It depends, as always, on what you're doing. There isn't a hard limit, but remember: an ID is a distinct task or thing. That thing should be quite granular, otherwise it's not a distinct thing: it's either more than one thing, or a category of things.

Let's revisit the 'repaint the kitchen' task. We decided that it deserved its own ID.

13.15 Paint the kitchen

So what might we save in this folder? Images of kitchens that we've researched online. The spreadsheet where we calculated the wall area and paint quantity. Colour samples. Paint receipts. Perhaps a dozen items?

⁸ <https://jdhq.johnnydecimal.com/support/about/naming-files/>

■ We are very organised

Remember, by the time we're at **13.15 Paint the kitchen** we are superbly organised. We are exactly where we need to be.

I think we've been conditioned – scared – by the folders we see at work. They contain dozens of items but it's a disaster. So we see a folder with dozens of items in it and instinctively think, 'chaos!'

It doesn't need to be this way. I think it's okay to have a folder containing a couple of dozen items.

But that's probably the limit. Much more than that and you should consider splitting it up. If you end up with dozens of artefacts after researching what colour the new kitchen should be, consider creating a new ID for the actual paint job.

44.02 You really create a folder to save one thing?

People wonder if I really create a *whole folder* to save just one thing. They think I'm mad. I think they're lazy. How much time does it take to create a folder? No time at all.

44.03 Saving files in subfolders

You should try to avoid creating folders in your Johnny.Decimal folders.

I used to say that you should *never* create subfolders, but my stance on this has softened. There are cases where it makes sense to create subfolders.

However. There's no point to any of this if you just go back to burying your files many layers deep. Stick to one level of subfolders, label them neatly, and don't have too many.⁹

⁹ Read more about neatly sorting subfolders using 'year-month' here <https://jdcml.al/11.06/>, and here <https://jdhq.johnnydecimal.com/support/about/subfolder-patterns/>

44.04 Numbers don't indicate order, but...

The numbers in Johnny.Decimal don't 'mean' anything. Something having a lower number doesn't mean it's more or less important: category 11 is no better than category 84.

That said, humans seek order. And so if there *is* a natural order to things, you might as well think about it now.

For example, if the areas you have designed correspond to a sequence in the real world it obviously makes sense to sequence your system in the same way. I hardly need tell you: you'd fight me if I told you *not* to do this.

But don't get hung up on this point. I've designed systems in the past and something has, later, felt 'out of order'. Other than being mildly annoying, it really doesn't matter.

44.05 Look for standards

The one exception to this is if you've identified any of your area/category combinations that are likely to become standards across your life.

The only example of this that I have is my 'project management' area. I'll show it here; forgive me if you're not a project manager but you'll get the idea.

If you *are* a project manager, this structure works with the PMP® system and should be immediately familiar to you. So familiar, in fact, that you should use it yourself.

This structure is always 'first' at area 10-19 and I'll type it here from memory.

10-19 Project management

- 10 Integration
- 11 Scope
- 12 Schedule
- 13 Cost
- 14 Quality
- 15 Resource
- 16 Communication
- 17 Risk
- 18 Procurement
- 19 Stakeholder

Furthermore, most of these things have a 'management plan'. It's the first thing you create, so guess what?

10-19 Project management

10 Integration

11 Scope

11.01 Scope management plan

12 Schedule

12.01 Schedule management plan

...

19 Stakeholder

19.01 Stakeholder management plan

I've used this across a few work projects, with the same manager.¹⁰ Imagine, as a manager, knowing where an artefact will be *before it's even been created*.

This is where the superpowers kick in. I told you the boss was going to love you.

45 Area complete

At this point you should have a working Johnny.Decimal system.

Your files are where they should be. You should be feeling calm.

We've focused on files here because they're the simplest and most common use-case for a Johnny.Decimal system. But in the next workbook section I'll explain how you can use your system to store everything else in your life.



¹⁰ As it happens, the same person that named Johnny.Decimal over a decade ago. Thanks, Alex!

50–59

The index/ JDex

Want more help? 

[Click here to watch this area in the online workshop!](#)

51 Overview

The Johnny.Decimal website contains an overview of the index (your JDex) at **11.05** and **12.02**: refresh your memory by re-reading those pages. We will cover all of the key points here in more detail.

<https://jdcm.al/11.05/> <https://jdcm.al/12.02/>

51.01 The index *is* your Johnny.Decimal system

Keeping an index is one of the things that people tend to push back on. They think it's extra work, which it is.

If this is how you feel, you haven't realised the pitfalls of *not* keeping an index, and the positive advantages of keeping one.

I consider the index to be more important than your file system. The index *is* your Johnny.Decimal system.

You need to be totally sold on this concept, so let's review before we go on.

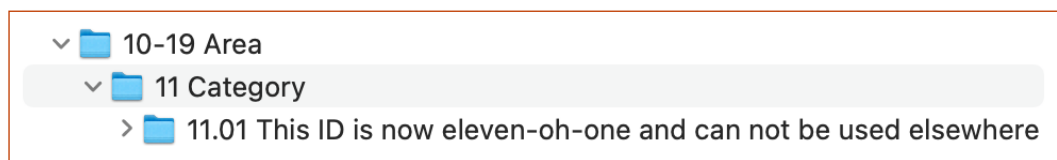
51.02 The pitfalls of not having an index

The Johnny.Decimal system relies on you being able to assign unique IDs to things. It would be a bit silly otherwise: imagine a system where you could use **11.01** to refer to any number of different things.

So without an index, how do you do this?

The most common response is that 'my file system is my index', meaning that the place that numbers are recorded – the one true place that defines what a Johnny.Decimal ID represents – is your file system.

In practice this means that 'creating a new ID' is equivalent to creating a file system folder. That ID is then 'used up', and its name is the name of the folder.



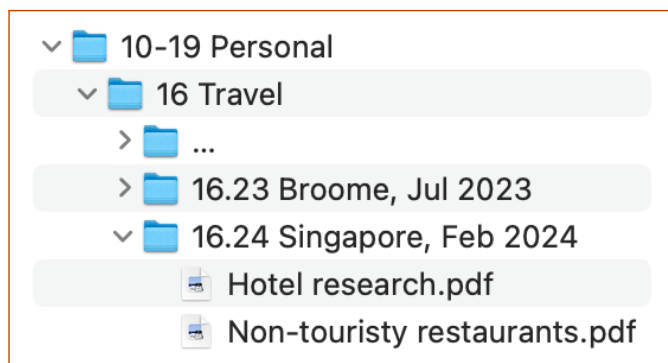
And this would work for 90% of people 90% of the time. You might like those odds, in which case please ignore my advice and use your file system. I wish you well.

I prefer certainty. Because here's what happens in reality.

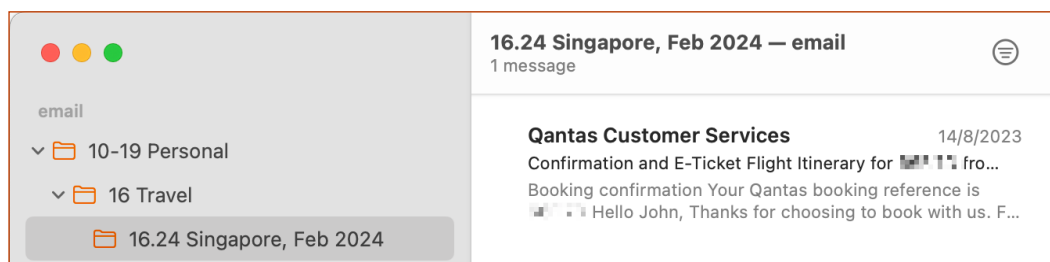
51.03 Case study 1: Organising travel

You're planning a holiday, and part of your research involves downloading brochures.

Of course you want to save them somewhere, so in your **Travel** category you create a new folder, assign the next available number, and start saving the PDFs that you download.



Later, you book the Singapore trip and want to save the email confirmation from the airline. You've already got an ID - **16.24** - so you create an email folder with that number and move the message there.

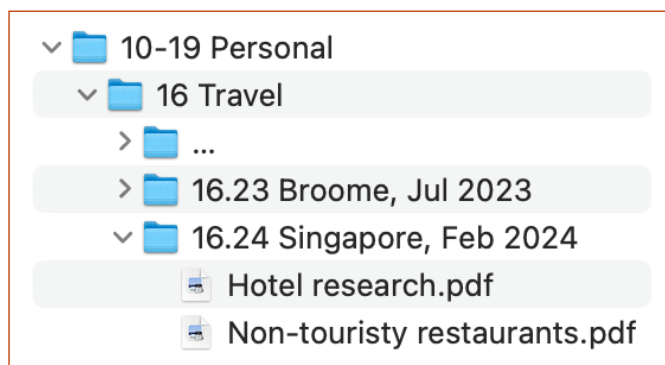
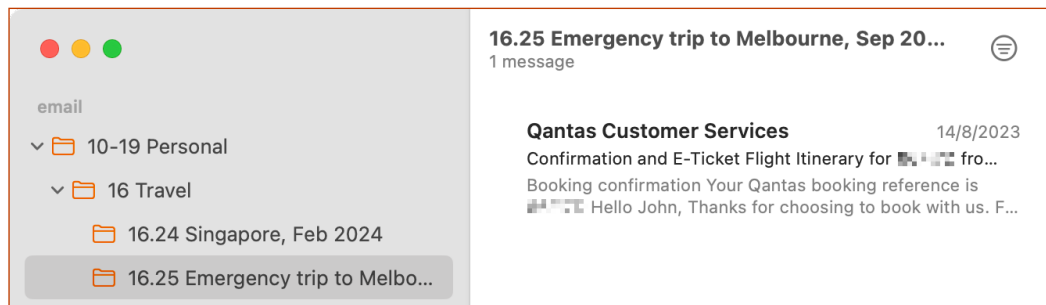


So far so good: you get a gold star.

Next, an impromptu trip interstate comes up. Family emergency, last minute. The only artefact that exists is the email confirmation and, by now giddy with the power that comes from being organised, you create a new folder in your email in which to save it.

The next number is obviously 16.25, so you use that and get on with life.

Here's the situation.



The next trip

Singapore was great and you decide to spend more time in Southeast Asia.

You start researching a holiday to Vietnam. You're downloading PDFs again, so you jump in to your file system to create a folder, and what do you do?



Now, reading this you're slapping your head and telling me that *of course* you wouldn't be so stupid to do this. *Of course* you somehow remember that you've already used 16.25 for that quick trip interstate, and *of course* you'll use 16.26 for the Vietnam holiday.

☒ You will not remember

I'm telling you now, you will not remember. Why do you think I realised that this system needed an index? This is something that I did myself, years ago.

This problem gets worse the more places you use your numbers.

As we'll see below, keeping notes using your Johnny.Decimal system is simple and powerful.

So now imagine that the *next* trip starts with a note, as you're just gathering ideas. Where do you find the number for that note? How do you remember that you've used it?

The answer is the index.

Fortunately we can combine our index and our notes system, so the 'wasted' overhead is eliminated.

51.04 Case study 2: Working as a consultant

Ah, my old life. By purchasing this workbook you've saved me from it, and I cannot thank you enough.

People who work for one company who are then farmed out to a 'customer' to do some work – consultants – usually have two computers.

There's the one for the company that employs you, and the one that your customer gives you to access their systems. Of course with these computers comes at least two file systems and two email accounts.

☒ One project, many physical systems

The problem is obvious. The *job* is one project. But the Johnny.Decimal system must account for items in two – sometimes more! – physical systems.

The index ties these physical systems together, and prevents overlap and re-use of numbers. There is one index, which might be synchronised across systems depending on how you implement it.

Each index entry 'points to' the location of the item as we'll see below.

51.05 Case study 3: Managing a team

I was the technical lead for the data centre migration project that I spoke about above. But we worked in a team, and it was important that everyone had access to the Johnny.Decimal system.

There's no point me keeping an organised shared folder structure if others are going to come in and do what they like.

The team created new artefacts and I needed them to be able to assign themselves numbers in the system. In reality, I managed the system and told them where to put things: but in an ideal world, people should be able to 'self-serve' their own Johnny.Decimal IDs.

At work, data exists all over the place. The index is the only way you have of centralising your project's knowledge.

I consider this an absolute truth, by the way. Whether or not you use Johnny.Decimal, I don't see how you could manage a project's data without an index of some sort; hence the chaos I saw at every place I ever worked.

52 Before you start

52.01 The various ways to keep your index

Here are the three primary ways that I recommend keeping your index.

The first two use text notes, and the third uses a database. I'll provide a quick overview here, and more detailed instructions below for each option.

You will select one of these methods; you do not need to be familiar with all three.

After reading this you might come up with another method. As long as it meets the needs of the index, that's great. If it's interesting, let me know.

52.02 Method 1: Individual notes, one per index entry

This method involves creating a new note for each index entry. That note *is* the index entry.

Note-keeping applications are common. You get one free with your Apple device: Apple's 'Notes'. There are other free and paid applications for all platforms. Search the internet for 'best notes app' and you'll find a bunch of lists.

I'll refer to this as your 'notes app' throughout this workbook.

The idea is as simple as 'create a note whose title is the Johnny.Decimal ID that you are allocating'.

☒ Advantages

- Quick, easy, and the most 'obvious' solution to a non-nerd.
- Keeping more data about your item is easy – for example where it is.
- As we'll see later, it is trivial to combine this method with your note-taking.

☒ Disadvantages

- Not as computer-readable. If you want to use scripts or other automation to manage your index, this option might not be for you.
- Depends on your access to a nice notes app. Frankly, the best ones are paid and exist on Apple's platforms. It just sucks to be a Windows user. ;-)
- Not as shareable, depending on which notes app you use. So this might not work if you're at work and you need the rest of the team to be able to access the index.
- Depending on which app you use, it can be difficult to see the overall structure of your Johnny.Decimal system.

52.03 Method 2: One note, each line being an index entry

Rather than using multiple notes, you can keep one single note. Its *contents* are your index.

We've already seen examples of this approach above. Whenever I've written out a Johnny.Decimal structure as text, this is essentially what I've done.

Using this method, you create a single note (or text file or document or spreadsheet or almost anything else) and the contents of that note are your index.

It looks something like this.

```
10-19 Red things
  11 Symbols & signs
    11.01 'Prohibited' in Japanese
    11.02 SOS
    11.03 Blood types
  12 Household things
    12.01 Telephone
    12.02 Drawing pin
    12.03 Lipstick
    12.04 Glass of red wine
  13 Sports equipment
    13.01 Boxing glove
    13.02 Table tennis bat
    13.03 Golf flag
  14 Red-haired people
    14.01 Jemima
    14.02 Simone
    14.03 Booker
    14.04 Nicole
20-29 Green things
  21 Food
    21.01 Apple
    21.02 Cucumber
    21.03 Kiwi fruit
  ...
```

There's slightly more to it, which we'll see below.

▣ Advantages

- Very computer-readable, especially if you're exact with your formatting. I provide a formal structure below at [55.06](#).
- If you store this in a location that allows collaborative editing – say, a Google Docs or Microsoft 365 document – it is easily shared with the team.

- If you're really adept at moving around large text files this approach may have benefits. (So if I tell you I'm writing this in Vim and you know what that means, this might be for you.)
- Not dependent on any particular notes app. Anything that reads text will work.
- Your entire Johnny.Decimal system is visible 'at-a-glance', so it's easy to understand its structure.

❏ Disadvantages

- I find the 'individual notes' method easier to search. But perhaps that's just what I'm used to: `cmd/ctrl-f` will search almost any text anywhere.
- Can become unwieldy with a large index.
- Less friendly for the beginner.
- Doesn't integrate your notes in to the index.

52.04 Method 3: A database

Rather than using notes, you can use an actual database. Databases are well suited to this task, but you have to know how to set up and maintain them.

Each of your IDs is a database record with associated data.

❏ Advantages

- Database columns can track metadata related to each index item.
- You could program an interface to allow easier creation of, and searching of, your index entries.
- You can query your database using any other program via its API.

❏ Disadvantages

- Not quite as immediate and easy in daily use. (Unless you build some sort of hotkey-activated front end, in which case call me immediately.)
- Now you're a database administrator.

53 The process

Whichever method you decide to follow, **please read the remaining sections first**, then record your choices in the 'my actions' worksheet.

We begin with the simplest method, but the concepts described there apply equally to the more complex methods. When you reach those I won't mention the concepts again – I'll assume you're an advanced user and you've got it.

53.01 Choose a method

I recommend and use the first method and include the other two for advanced users. If none of this means much to you, just choose the first one – individual notes, one per index entry – and ignore the others.

▣ Which notes app?

If you choose methods 1 or 2, you'll need a notes app to keep your notes in. Apple Notes will do the job. I prefer Bear¹¹ for Apple's platforms. Obsidian¹² is a popular choice for other platforms.

I will keep this generic and not demonstrate the specific features of any one notes app. What Bear or Obsidian can or can't do isn't the point here. All screenshots show the application as it comes 'out of the box'.

You may already have a notes app that you like. If not, once you have chosen one spend some time getting to know it.

If you need help choosing a notes app, ask us in the forum¹³.

Worksheet action

Action ID : 53.01

¹¹ <https://bear.app/>

¹² <https://obsidian.md/>

¹³ <https://forum.johnnydecimal.com/>

54 Method 1: Individual notes, one per index entry

With this method, the most important attribute of your chosen notes app is its ability to find and create notes.

This might sound obvious, but there's a big difference between an app whose search feature exists but isn't very good, and one whose search feature is top-notch.

I consider Apple Notes' search to be okay. It works. It finds things. But it's sluggish and the results tend to jump around a bit.

Bear's search is amazing. It's *much* quicker and just more pleasant to use. Sure, Bear is US\$30 a year.¹⁴ But when you're using it all day, the cost is easily worth it.

54.01 Basics: create one note for each Johnny.Decimal ID

In the previous workbook area you created your Johnny.Decimal structure in your file system. Let's migrate that system to your new notes app.

We're not moving any of your files: we're just moving the *management* of your numbers, which currently sits with your file system, to your notes app.

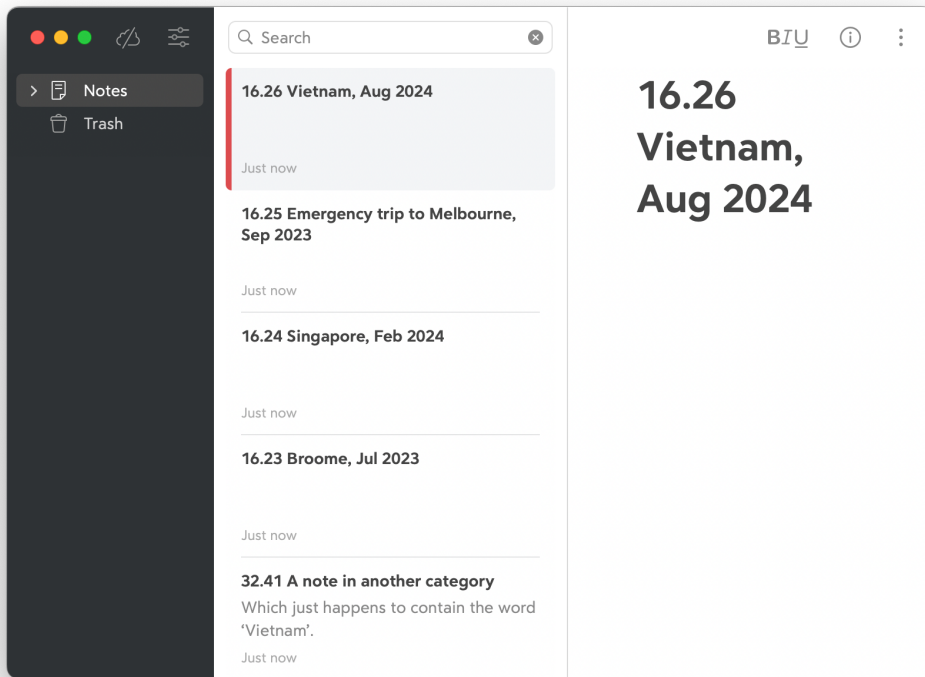
To do this, just **create a new note for each of the Johnny.Decimal IDs that you created above**. That's it: at this point, the note can just contain the title of the item.

Mine always start with the Johnny.Decimal ID, and then the title of the item. I don't use a colon or a dash or any other symbol between the ID and the title.

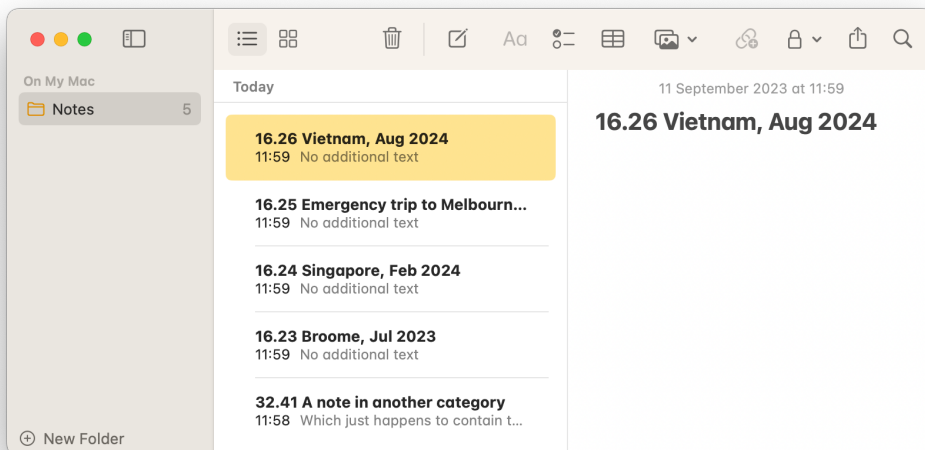
16.26 Vietnam, Aug 2024

The main concern is that your list of notes – either unfiltered or filtered with search – must clearly show the Johnny.Decimal ID and title.

¹⁴ Bear is free to download and use. If you want the advanced features such as cloud sync, which you do, it's US\$30 a year.



Bear



Apple Notes

54.02 Start somewhere clean

It's nice to start somewhere 'clean'. Don't mix up your shiny new index with a bunch of old crud.

If you've just downloaded Bear and there's nothing else in it, great.

Otherwise have a look and see if your notes app has the concept of folders. For example, in Apple Notes you can create a new folder in the left pane.

54.03 Searching for your items

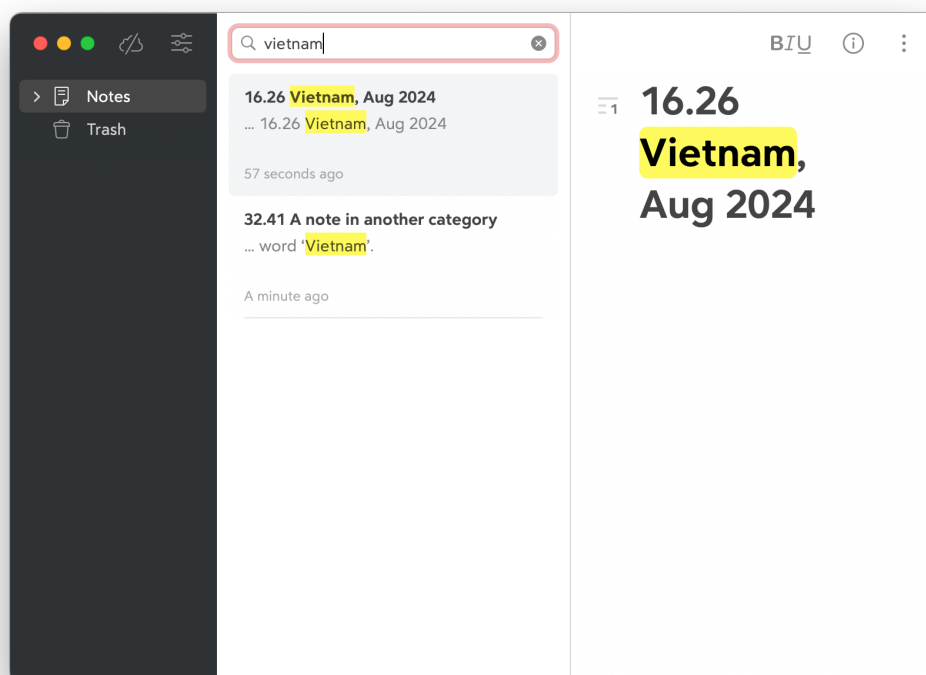
This part is crucial, and it's where you need to do a little mental flip.

Right now, if I were to ask you to find a thing, what would you do? I'm betting your first instinct is to head to your file system. Because most things are files, right?

I need you to switch this behaviour. It'll take time. Your first instinct instead should be to reach for your index.

Let's find that trip to Vietnam. Where do you search for it? **You search your index.**

What does your index tell you? Let's have a look.



Search behaves essentially the same for each app, so rather than showing you three screenshots each time I'll just include my favourite, Bear.

At its most basic, your index tells you that this item exists. As there are no other notes about Vietnam in your notes app, this information is probably in your file system.

This might feel silly. You just used your notes app to look up something on your file system. And in this trivial setup it *is* a bit silly, but stay with me and I'll show you where it gets useful.

54.04 Remember that emergency trip?

Here's where it gets useful. That was quick.

Had you gone looking for that emergency family trip in your file system then you would *not* have found it: it only exists in your email.

You would have become confused. You know the trip exists. Where is it?

And then you realise – *aah, it's in my email!*

This confusion is real. It causes us to lose focus. It causes us to resent the computer; rather than sitting down and facing a tranquil ocean, we face a raging sea. This is turning in to a self-help book but it's true! Your computer needs to be a happy place.

The index makes it so. If you had consulted your index first, you would have been directed to your email with no fuss.

54.05 Sorting your items

Your notes app almost certainly has two ways to sort the list of notes: alphabetically, or by date modified or date created.

I switch between these views depending on what I'm doing.

Alphabetically (or *reverse* alphabetically so that 16.26 is above 16.25) is 'neat', but you might find that sorting by date modified is more useful day-to-day as the notes you just edited will be at the top of the list.

54.06 Advanced search

In the basic search for 'vietnam' shown above, we've actually matched two notes: the one we want, and another which just happens to contain the word 'Vietnam'.

This is sometimes what we want: the search feature in these apps searches *everything*, which is very powerful. It means that you can mention something anywhere in any note and be able to find it.

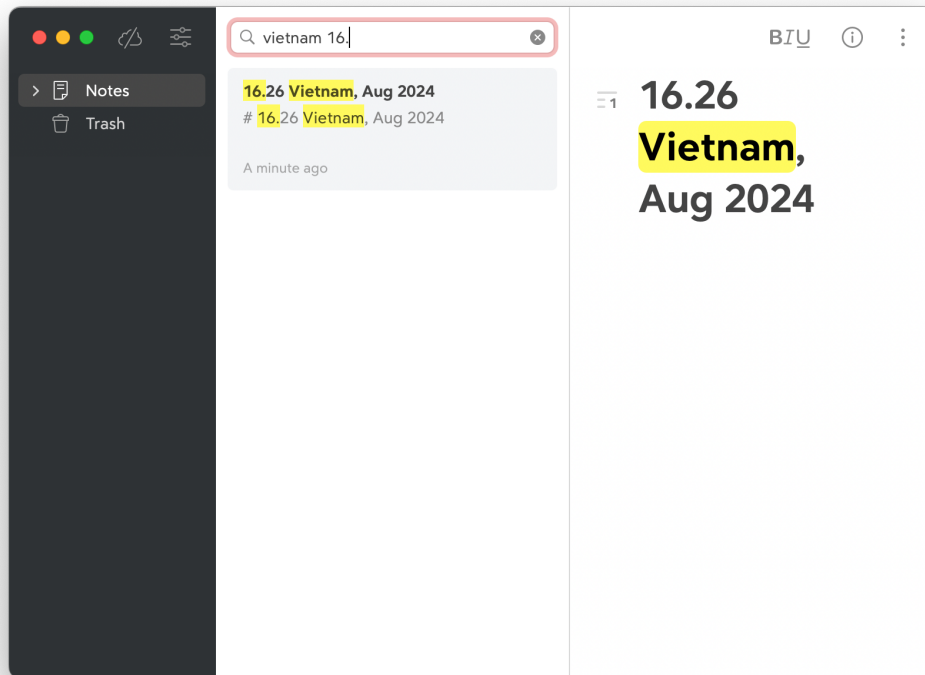
But if you have a lot of notes, the extra results can get in your way. This is where we use Johnny.Decimal to our advantage.

🔍 Use the Decimal point, Luke

You know that 16 is your Travel category, so to narrow your search to only that category, simply add it to the search.

Adding 16., including the decimal point, gives you the best chance of finding what you want. Because *all* of your travel items contain the text 16.: it's right there in the title of the note!

The characters 16 might appear in the middle of a phone number or an address in another note. But 16. is less likely to result in a false positive.



54.07 Use this to browse your categories

If we *start* our search with **16.**, the effect is to instantly filter our notes to only show items in category **16 Travel**.

When my system grows – say I’m a year in to a job and I have hundreds of items – this is usually how I begin my search.

The default settings of your notes app probably show the most recently edited notes at the top of the list, so you might find that this is all you need. The note you’re looking for is often right there.

54.08 Recording the location of your items

We’ve already seen that there are various places that you can store your stuff.

- Documents in your file system.
- Notes in your notes app.
- Emails in your email client.

If you're at work it's worse: you have multiple file systems, document management systems, chat threads, SharePoint libraries. There are literally dozens of places a thing could be.

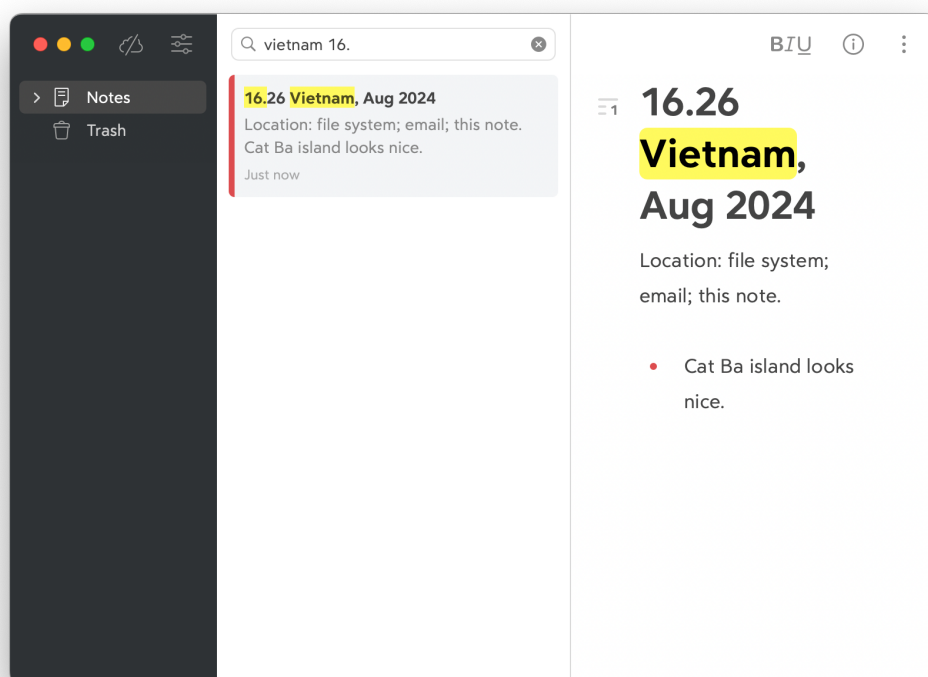
Keep it simple

We use the index to tell ourselves where our items are. It's as simple as typing out the words.

I always put the location immediately after the title. Here, we're indicating that we've stored something in our file system, our email, and in this note.

Because the Johnny.Decimal number tells us all that we need to know – it's enough for us to navigate to the folder in our file system – you don't need to be any more prescriptive.

The beauty of this is that it's just words. If you don't like this particular notation, invent your own. **All you're doing is telling yourself where a thing is.**



54.09 Recording the structure of your system

The disadvantage of this method is that it does not show the overall structure of your system. We 'know' that 16 Travel is a category, but so far we have not written that down anywhere.

Given that creation of areas and categories is rare, we can solve this by creating a note with the structure of our system. We shouldn't need to update it very often.

Call the note **00.00 Index** and add **only your area and category structure**.

```
10-19 Red things
  11 Symbols & signs
  12 Household things
  13 Sports equipment
  14 Red-haired people

20-29 Green things
  21 Food
  22 Animals
  23 Plants

30-39 Blue things
  31 Emoji
  32 Symbols
  33 Animals
  34 Clothing
```

If you're using Apple Notes, you can create folders and subfolders to show your structure in the sidebar. If you're using Bear you can use its tags feature¹⁵ to do this. This is outside the scope of this document and is left as an exercise for the reader.

Hint: nest your tags like
#10-19 Red things/11 Symbols & signs#
and don't forget that closing hash.

In the future, Johnny.Decimal University¹⁶ will have a course explaining how I use all of Bear's features to manage my JDex.

54.10 Disambiguate items with notes

Ambiguity is inevitable. This system *helps*, but I don't ever claim that it's a panacea.

¹⁵ <https://bear.app/faq/how-to-use-tags-in-bear/>

¹⁶ <https://jdcm.al/14.02/>

Help your future self as much as possible. If you misplace a physical object, common advice is (after finding it) to store it in the place you first looked for it. Because that's where your brain expects it to be.

You can use your index the same way.

Here's an example. I did this yesterday.

Generating 'dummy' Johnny.Decimal files

I need to generate hundreds of Johnny.Decimal index entries that aren't my real index entries. Otherwise when I record my screen to demonstrate a concept I'm conscious that I'm leaking personal information.

I wrote a short script using Deno, a programming language, and need somewhere to store it. Here's where I chose:

D85 Johnny.Decimal business	(system)
50-59 Creative	(area)
54 Sample JD systems	(category)
54.06 Deno script to generate JD index files	(ID)

This is one of those things that I worry about losing in the future, because it could fit in many of my mental buckets. It's code, and I have another place where I store code. Finding it here depends on me remembering that I wrote it to create a 'Sample JD system (54)', and in a few months I might not do that.

I even have to remember that it was done for the 'Johnny.Decimal business (D85)' and not the website (D01) or as part of a personal project (P01).

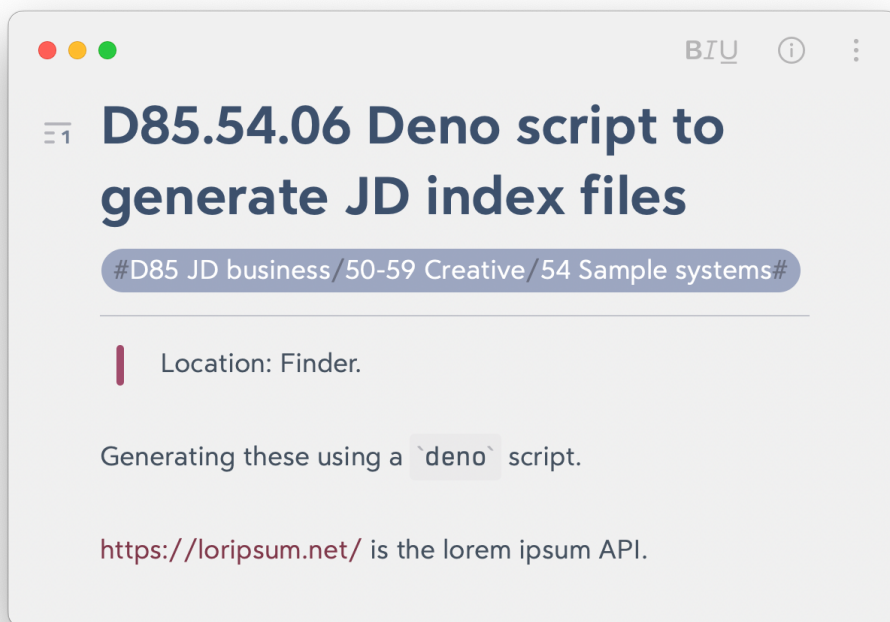
So what *will* I remember?

What will I absolutely, without fail, remember about this in a few months?

- That I wrote it using Deno, because it's the first time I've done that.
- That I used a 'lorem ipsum' generator, and
- That I used the lorem ipsum generator's API.

(The details of what all this means don't actually matter.)

So here's the index entry for this ID.



Typing those extra words took ten seconds. But now any combination of words that includes 'deno', 'lorem ipsum', and 'api' will hit this – and currently no other – index entry.

I tested my search after making the note, assuming that 'lorem ipsum' was in the URL I pasted. It isn't! So I added '... is the lorem ipsum API'.

Words are free. Type more words.

55 Method 2: One note, each line being an index entry

With this method, it doesn't really matter which notes app you use.

You won't be interacting with it much: everything will be in one single text file. You'll be using the standard 'find-text-within-the-document' feature of whatever app you use to view the file.

This method is so simple that you can use the notes app – they were called 'text editors' back then – that came with your computer.

On a Mac this is TextEdit and on Windows we have trusty Notepad. Both work admirably well for this purpose.

To create your index using this method, simply **type out your index as shown** throughout this document.



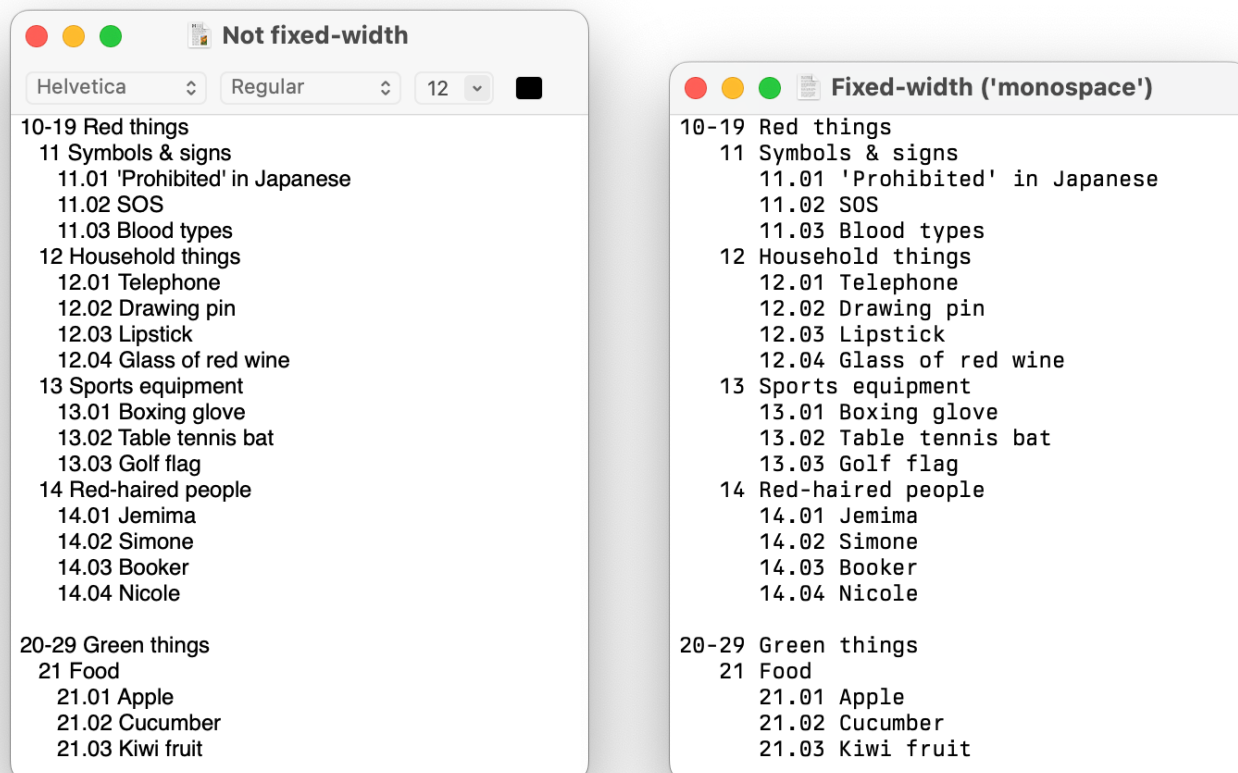
55.01 This file is ID 00.00

Of course this file has to be saved somewhere in your Johnny.Decimal system. I always reserve **00 Index** as a category, and this note is **00.00 Index**.

Think of it as the very beginning of your system. It's the place you go first, in order to find out where you need to be.

55.02 Fixed-width fonts work better

I prefer fixed-width fonts for this purpose. If your font isn't fixed-width, the numbers won't line up.



Your operating system comes with a few, listed below. You can search the web for more, but beware: this is an easy rabbit-hole to fall down.

Mac

- Andale Mono
- Menlo
- Monaco

Windows

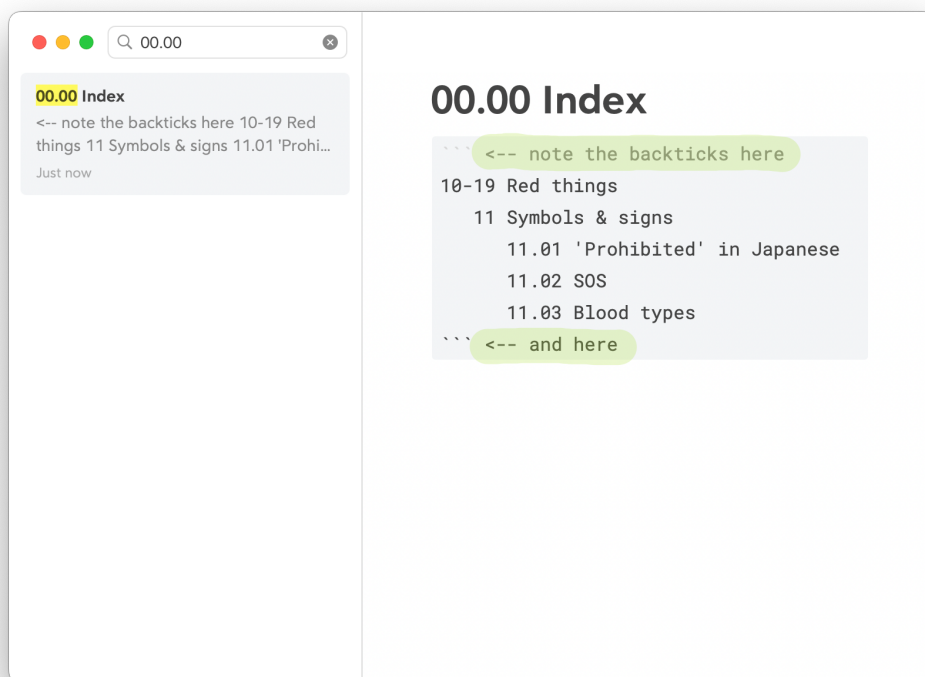
- Consolas
- Courier

If you're using TextEdit, to make it look like the screenshot on the right (above) select *Format > Make Plain Text* from the menu. Notepad gives you a fixed-width view by default.

55.03 Look for a 'code block' feature

Many notes apps have a feature that turns a block of text in to a 'code block'. Code is formatted using a fixed-width font, and this is what we need.

Look for it in your editor's menus or help system, or you can try entering a block of text between three backticks (usually above the *Tab* key).¹⁷

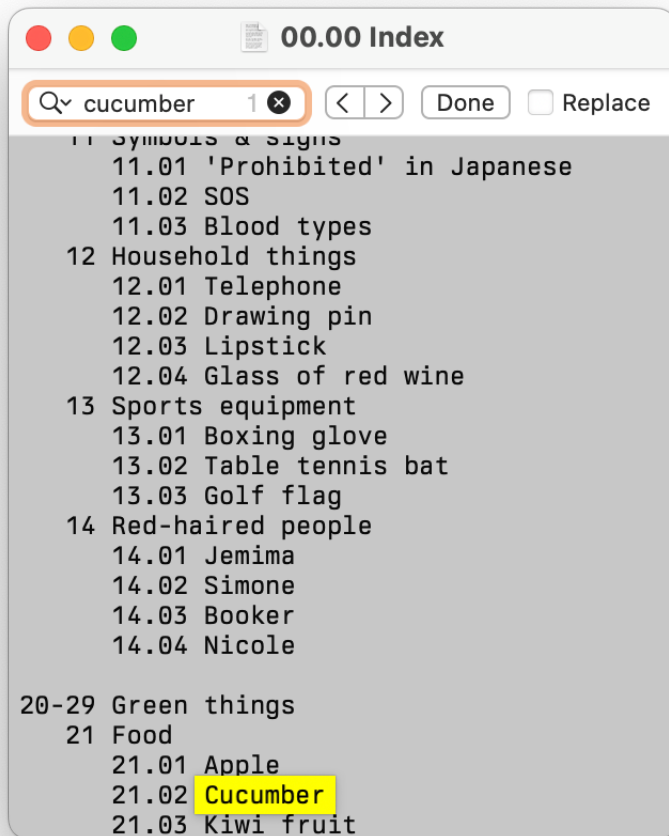


If you're using Apple Notes format the text as *monospaced*.

55.04 Searching for your items

Searching for your items using the one-file method is as simple as using the 'find' feature of your notes app or text editor.

¹⁷ But it can move around depending on keyboard layout (e.g. in the UK it's often next to the 'z').



I use the term 'find' vs. 'search' here because it's subtly different.

🔍 *Searching for a note*

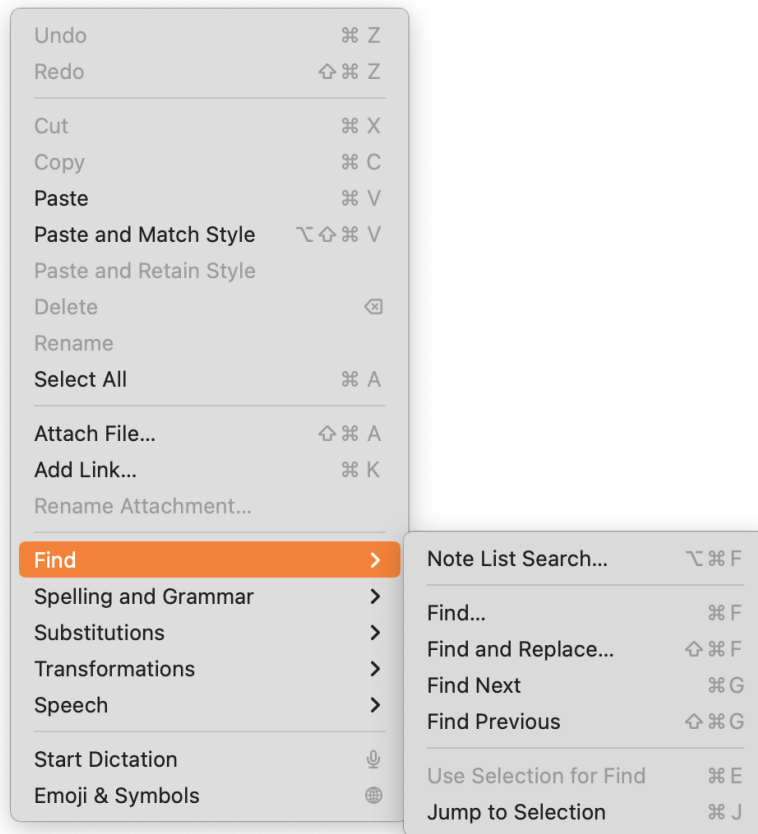
- is searching your many notes for the one you want.

🔍 *Finding within a note*

- is finding the particular text, in that note, that you're looking for. It's this action that we use in the one-note scenario.

Your notes app will have different keyboard shortcuts for these two actions.

For example, here's Apple Notes' *Edit* menu. You can perform a *Note List Search* (which searches all of your notes), or *Find* text within the current note.



Play around with your notes app until you're really comfortable with these actions.

55.05 Recording the location of your items

When using the one-file method, we add 'metadata' – data about our data – below each entry.

```
10-19 Red things
  11 Symbols & signs
    11.01 'Prohibited' in Japanese
      - Location: email.          <-- new line
    11.02 SOS
      - Location: file system.    <-- new line
    11.03 Blood types
  12 Household things
    12.01 Telephone
    12.02 Drawing pin
    ...
```

This syntax can be used to store any data related to the index entry.

This will be most useful to you if you want to 'parse' your index using a computer program. You can store any key/value pair this way.

If this doesn't mean much to you, you can safely ignore it. (You should probably be using method 1.)

On the line directly below the index entry, enter:

- a dash,
- a space,
- the name of the data key,
- a colon,
- a space,
- the value of the data.

(As shown above.)

Whitespace (i.e. leading space) should be ignored by your parser, but displayed for legibility.

55.06 Formal JD spec

If you're keeping a single file as your index, it makes sense for the content to be consistently formatted.

One of the reasons you'd do this would be so that you could query it programmatically. That only works if the format is well-defined.

This is the formal specification for the Johnny.Decimal index file. (I need to properly formalise it, but this is a good start.)

<https://github.com/johnnydecimal/index-spec>

56 Method 3: A database

If you know what a database is and how you'd get yourself one, you don't need me to say much more.

Airtable¹⁸ is a great SaaS database. (That said, I wish its interface was a bit quicker. And that they'd stop putting the price up.)

I've created a template database which you can use as a starter. You can copy it from here.¹⁹ Click the 'Copy base' button in the menu bar to copy it to your own account.

I don't currently use this database myself so it may need a bit of work.

56.01 SharePoint Lists

If you're at work, you might have access to SharePoint. It comes with a database feature: just create a list.²⁰

They work fairly well and are shared by default.

57 Area complete



¹⁸ <https://airtable.com/>

¹⁹ <https://airtable.com/app2QDvkeHs28xIgm/shr5vvlmaCV98G6z7>

²⁰ <https://jdc.m.al/d41.56.01/>



90–99

Appendix

91 Licence

This work is licenced under a Creative Commons Attribution-NonCommercial 4.0 International Licence.

You are free to share and adapt this work for non-commercial purposes, as long as you give appropriate credit.

View the full licence here: <https://creativecommons.org/licenses/by-nc/4.0/>

92 Copyright

This document is copyright 2025 John Noble <hello@johnnydecimal.com>.

93 Worksheets

93.01 My actions

Page 89.

93.02 My system's scope

Page 91.

93.03 My areas and categories

Page 92.

00-09 About this workbook

Started on : yy / mm / dd

- | | | |
|-------|-----|--|
| 01.03 | [] | I've skimmed through the whole workbook so I know what's coming. |
| 01.05 | [] | I understand the concept of Johnny.Decimal areas, categories, and IDs. |
| 02.02 | [] | If I need one, I have assigned an 'emergency storage location' for my files while I complete this workbook.

That location is: |

Completed on : yy / mm / dd

10-19 Define your system's scope

Started on : yy / mm / dd

- | | | |
|-------|-----|--|
| 13.02 | [] | I have defined my system's scope.

Use the 'my system's scope' worksheet at 93.02 and keep it to hand as you complete the next area. |
|-------|-----|--|

Completed on : yy / mm / dd

20-29 Discovery

Started on : yy / mm / dd

- | | | |
|-------|-----|---|
| 23.01 | [] | I have spent a week writing everything I do, see, think, have, want, and need, on sticky notes or in a mind map.

I have resisted the temptation to organise these notes.

I am comfortable that I have captured all of the elements of this 'system' before I move on. |
| 24.07 | [] | If there are other people who will use this 'system' – my partner, family, friends, colleagues – I have asked them for their input. |

Completed on : yy / mm / dd

30-39 Create your areas & categories

Started on : yy / mm / dd

- | | | |
|-------|-----|---|
| 33.04 | [] | I have grouped all my notes in to areas and categories, ideally trying it both ways: areas then categories, and categories then areas.

My area and category names have been recorded on the 'my areas and categories' worksheet at 93.03 . |
|-------|-----|---|

Completed on : yy / mm / dd

40-49 Build your system

Started on : yy / mm / dd

- | | | |
|-------|-----|--|
| 43.03 | [] | I have assigned numbers to my areas, categories, and IDs. |
| 43.04 | [] | I have created folders in my file system and moved my files in to those folders. |

Completed on : yy / mm / dd

50-59 The index/JDex

Started on : yy / mm / dd

- | | | |
|-------|-----|---|
| 53.01 | | I have chosen the following method to keep my JDex: |
| | [] | Method 1: Individual notes, one per index entry

My notes app is: |
| | [] | Method 2: One note, each line being an index entry

I have created a file called 00.00 Index.txt to record the structure of my system. |
| | [] | Method 3: A database

I am on my own because databases are complicated. :-) |

Completed on : yy / mm / dd

System name: _____

Date: _____

Goals for my system _____

In scope for my system _____

Not sure about these yet _____

~~~~~

Not in scope for my system \_\_\_\_\_

| Area name | Names of <i>at most</i> 10 categories in this area |  |  |  |  |  |
|-----------|----------------------------------------------------|--|--|--|--|--|
| 1         |                                                    |  |  |  |  |  |
|           |                                                    |  |  |  |  |  |

| Area name | Names of <i>at most</i> 10 categories in this area |  |  |  |  |  |
|-----------|----------------------------------------------------|--|--|--|--|--|
| 2         |                                                    |  |  |  |  |  |
|           |                                                    |  |  |  |  |  |

| Area name | Names of <i>at most</i> 10 categories in this area |  |  |  |  |  |
|-----------|----------------------------------------------------|--|--|--|--|--|
| 3         |                                                    |  |  |  |  |  |
|           |                                                    |  |  |  |  |  |

| Area name | Names of <i>at most</i> 10 categories in this area |  |  |  |  |  |
|-----------|----------------------------------------------------|--|--|--|--|--|
| 4         |                                                    |  |  |  |  |  |
|           |                                                    |  |  |  |  |  |

| Area name | Names of <i>at most</i> 10 categories in this area |  |  |  |  |  |
|-----------|----------------------------------------------------|--|--|--|--|--|
| 5         |                                                    |  |  |  |  |  |
|           |                                                    |  |  |  |  |  |

Do not try to 'use' all of your areas and categories

Do not try to 'use' all of your areas and categories

| Area name |  | Names of <i>at most</i> 10 categories in this area |  |  |  |  |  |
|-----------|--|----------------------------------------------------|--|--|--|--|--|
| 6         |  |                                                    |  |  |  |  |  |
|           |  |                                                    |  |  |  |  |  |

| Area name |  | Names of <i>at most</i> 10 categories in this area |  |  |  |  |  |
|-----------|--|----------------------------------------------------|--|--|--|--|--|
| 7         |  |                                                    |  |  |  |  |  |
|           |  |                                                    |  |  |  |  |  |

| Area name |  | Names of <i>at most</i> 10 categories in this area |  |  |  |  |  |
|-----------|--|----------------------------------------------------|--|--|--|--|--|
| 8         |  |                                                    |  |  |  |  |  |
|           |  |                                                    |  |  |  |  |  |

| Area name |  | Names of <i>at most</i> 10 categories in this area |  |  |  |  |  |
|-----------|--|----------------------------------------------------|--|--|--|--|--|
| 9         |  |                                                    |  |  |  |  |  |
|           |  |                                                    |  |  |  |  |  |

| Area name |  | Names of <i>at most</i> 10 categories in this area |  |  |  |  |  |
|-----------|--|----------------------------------------------------|--|--|--|--|--|
| 0         |  |                                                    |  |  |  |  |  |
|           |  |                                                    |  |  |  |  |  |

(Try to reserve area 00-09 for the 'System'.)

---

## 99 Index

|                                                         |           |
|---------------------------------------------------------|-----------|
| <b>00-09 About this workbook</b>                        | <b>5</b>  |
| 01 Overview                                             | 6         |
| 01.01 How to use this workbook                          | 6         |
| 01.02 Printing                                          | 6         |
| 01.03 Read it through before you start                  | 6         |
| 01.04 What was that box?                                | 7         |
| 01.05 What is Johnny.Decimal?                           | 7         |
| 01.06 Help! I'm stuck                                   | 7         |
| 01.07 Definitions                                       | 8         |
| 02 Before you start                                     | 8         |
| 02.01 How long will this take?                          | 8         |
| 02.02 Emergency! I have stuff I need to organise now    | 9         |
| 02.03 Help me convince other people                     | 10        |
| 02.04 This is a hard problem                            | 11        |
| 03 How to get more help                                 | 11        |
| 03.01 Our websites                                      | 11        |
| 03.02 Johnny.Decimal University                         | 12        |
| 03.03 Life Admin System                                 | 12        |
| 03.04 Small Business System                             | 12        |
| 03.05 Join our supportive (free!) community             | 12        |
| 03.06 Stay in touch                                     | 12        |
| <b>10-19 Define your system's scope</b>                 | <b>13</b> |
| 11 Overview                                             | 14        |
| 12 Before you start                                     | 14        |
| 12.01 How small can a system be?                        | 14        |
| 12.02 How big can a system be?                          | 15        |
| 12.03 Painting vs. remodelling vs. rebuilding           | 15        |
| 12.04 Similar, smaller projects: the 'creative' pattern | 17        |
| 12.05 Don't think about areas and categories yet        | 18        |
| 13 The process                                          | 19        |
| 13.01 System scope                                      | 19        |
| 13.02 Define your system's scope                        | 21        |
| 14 As you complete this process                         | 22        |
| 14.01 Go for walks                                      | 22        |
| 14.02 Use a pencil                                      | 22        |
| 15 Area complete                                        | 22        |
| <b>20-29 Discovery</b>                                  | <b>23</b> |
| 21 Overview                                             | 24        |
| 22 Before you start                                     | 24        |
| 22.01 Sticky notes vs. digital                          | 24        |
| 22.02 So what do I write on these sticky notes?         | 25        |
| 22.03 Not just your files                               | 26        |
| 22.04 Check your notepad                                | 26        |
| 22.05 Pay attention in meetings                         | 27        |

|                                                          |           |
|----------------------------------------------------------|-----------|
| 22.06 I'm not running 'a work project'                   | 27        |
| 22.07 Consolidate your files (if you want)               | 28        |
| 22.08 I just started a big new thing, what do I do?      | 28        |
| 23 The process                                           | 28        |
| 23.01 Spend a week writing sticky notes                  | 28        |
| 23.02 Using a mind map                                   | 29        |
| 24 As you complete this process                          | 30        |
| 24.01 Refer to your scope statement                      | 30        |
| 24.02 How many notes do you expect me to write?          | 30        |
| 24.03 You're really expecting me to do this for a week?  | 31        |
| 24.04 Can I...                                           | 31        |
| 24.05 Stare at your sticky notes sometimes               | 31        |
| 24.06 Use your experience                                | 31        |
| 24.07 Engage the team                                    | 32        |
| 25 Area complete                                         | 32        |
| <b>30-39 Create your areas &amp; categories</b>          | <b>33</b> |
| 31 Overview                                              | 34        |
| 32 Before you start                                      | 34        |
| 32.01 The Johnny.Decimal structure and its constraints   | 34        |
| 32.02 What's an 'area'?                                  | 35        |
| 32.03 What about 'category'?                             | 35        |
| 33 The process                                           | 36        |
| 33.01 Group your sticky notes                            | 36        |
| 33.02 Do I do my areas or my categories first?           | 36        |
| 33.03 Home > Insurance, or Insurance > Home?             | 37        |
| 33.04 Name your areas and categories                     | 37        |
| 34 As you complete this process                          | 38        |
| 34.01 How many categories should I have?                 | 38        |
| 34.02 Divide and conquer                                 | 38        |
| 34.03 How many things should each category contain?      | 39        |
| 34.04 When you're in a category, you're organised enough | 39        |
| 34.05 Is one hundred enough things?                      | 40        |
| 34.06 Look for ways to create standards                  | 41        |
| 34.07 Dealing with conflict                              | 41        |
| 34.08 Filename length limitations                        | 41        |
| 34.09 You're performing a 'card sort'                    | 42        |
| 35 Area complete                                         | 42        |
| 35.01 Example: if your job was to organise emoji         | 42        |
| 35.02 Emoji by category then area                        | 45        |
| <b>40-49 Build your system</b>                           | <b>47</b> |
| 41 Overview                                              | 48        |
| 42 Before you start                                      | 48        |
| 42.01 Johnny.Decimal: a refresher                        | 48        |
| 42.02 Areas and categories are containers                | 49        |
| 42.03 IDs are where we store our stuff                   | 51        |

|              |                                                    |           |
|--------------|----------------------------------------------------|-----------|
| 43           | The process                                        | 52        |
| 43.01        | Assign numbers to your areas and categories        | 52        |
| 43.02        | Time to think again                                | 53        |
| 43.03        | When you're sure, create your IDs                  | 53        |
| 43.04        | Move your files                                    | 55        |
| 44           | As you complete this process                       | 55        |
| 44.01        | How many things do I save in each folder?          | 55        |
| 44.02        | You really create a folder to save one thing?      | 56        |
| 44.03        | Saving files in subfolders                         | 56        |
| 44.04        | Numbers don't indicate order, but...               | 57        |
| 44.05        | Look for standards                                 | 57        |
| 45           | Area complete                                      | 58        |
| <b>50-59</b> | <b>The index/JDex</b>                              | <b>59</b> |
| 51           | Overview                                           | 60        |
| 51.01        | The index is your Johnny.Decimal system            | 60        |
| 51.02        | The pitfalls of not having an index                | 60        |
| 51.03        | Case study 1: Organising travel                    | 61        |
| 51.04        | Case study 2: Working as a consultant              | 63        |
| 51.05        | Case study 3: Managing a team                      | 64        |
| 52           | Before you start                                   | 64        |
| 52.01        | The various ways to keep your index                | 64        |
| 52.02        | Method 1: Individual notes, one per index entry    | 65        |
| 52.03        | Method 2: One note, each line being an index entry | 66        |
| 52.04        | Method 3: A database                               | 67        |
| 53           | The process                                        | 68        |
| 53.01        | Choose a method                                    | 68        |
| 54           | Method 1: Individual notes, one per index entry    | 69        |
| 54.01        | Basics: create one note for each Johnny.Decimal ID | 69        |
| 54.02        | Start somewhere clean                              | 71        |
| 54.03        | Searching for your items                           | 71        |
| 54.04        | Remember that emergency trip?                      | 72        |
| 54.05        | Sorting your items                                 | 73        |
| 54.06        | Advanced search                                    | 73        |
| 54.07        | Use this to browse your categories                 | 74        |
| 54.08        | Recording the location of your items               | 74        |
| 54.09        | Recording the structure of your system             | 75        |
| 54.10        | Disambiguate items with notes                      | 76        |
| 55           | Method 2: One note, each line being an index entry | 78        |
| 55.01        | This file is ID 00.00                              | 79        |
| 55.02        | Fixed-width fonts work better                      | 80        |
| 55.03        | Look for a 'code block' feature                    | 81        |
| 55.04        | Searching for your items                           | 81        |
| 55.05        | Recording the location of your items               | 83        |
| 55.06        | Formal JD spec                                     | 84        |
| 56           | Method 3: A database                               | 84        |
| 56.01        | SharePoint Lists                                   | 85        |

|                               |           |
|-------------------------------|-----------|
| 57 Area complete              | 85        |
| <b>90-99 Appendix</b>         | <b>87</b> |
| 91 Licence                    | 88        |
| 92 Copyright                  | 88        |
| 93 Worksheets                 | 88        |
| 93.01 My actions              | 88        |
| 93.02 My system's scope       | 88        |
| 93.03 My areas and categories | 88        |
| 99 Index                      | 94        |



# johnnydecimal.com

A system to organise your life