Transcript: How I Built a 7,000-Listing Directory Website in 50 Minutes (Using Only AI Tools)

Video ID: opB25teOxYQ

Extraction Date: 2025-04-03 02:02:40

**[00:00:00]** today in this video I'm going to show

**[00:00:01]** you how I built montor find.com in just

**[00:00:04]** 7 Days basically it is a directory

**[00:00:07]** website that will be SEO optimized and

**[00:00:10]** it has 7,000 listings thereabouts and I

**[00:00:13]** scraped a whole bunch of data from the

**[00:00:15]** internet and I did this all basically

**[00:00:17]** with AI tools turned around a pretty

**[00:00:19]** nice little website so I think anyone

**[00:00:21]** can do this these days if you're

**[00:00:22]** remotely competent online I mean if you

**[00:00:24]** even know what SEO means I'm sure you're

**[00:00:26]** competent enough to figure this out

**[00:00:28]** we're going to use cursor Ai and bolt.

**[00:00:30]** new so without further Ado let's get

**[00:00:32]** started first I'll just show you the

**[00:00:34]** website and what we're looking at

**[00:00:35]** basically you can go to any state in the

**[00:00:37]** Union and you can find a Monas School a

**[00:00:41]** list of Monas schools as you can see it

**[00:00:43]** has the the name of the city the name of

**[00:00:45]** the State uh and then we've got reviews

**[00:00:47]** that are scraped from Google Maps so

**[00:00:48]** here's a four a five star review um out

**[00:00:51]** of four total I might have to clean that

**[00:00:52]** up it's got a map embed here with the

**[00:00:54]** proper instructions it's got a link to

**[00:00:56]** Google Maps and it's got phone number

**[00:00:59]** it's got it so all this these

**[00:01:01]** descriptions in every single uh listing

**[00:01:05]** were scraped with python script open

**[00:01:07]** source called crawl for AI which we're

**[00:01:09]** going to get into and then an llm an

**[00:01:11]** open source llm was used to make these

**[00:01:13]** descriptions um pretty fun project

**[00:01:16]** overall here's the project window in

**[00:01:18]** cursor as you can see we've got a big

**[00:01:21]** Json file full of all the listings here

**[00:01:23]** which will converted from a um from a

**[00:01:26]** CSV we've got a Blog in here as well um

**[00:01:29]** and a pretty fairly easy system for

**[00:01:31]** making new SEO blog posts and then

**[00:01:34]** there's a frequently asked questions

**[00:01:36]** which in and of itself I think is a bit

**[00:01:38]** of a masterpiece though I can't take

**[00:01:40]** full credit for it because Claude AI

**[00:01:42]** really helped me the first thing you

**[00:01:44]** need to do which I won't be covering in

**[00:01:45]** this video is find your lucrative Niche

**[00:01:47]** find the right search term which doesn't

**[00:01:49]** have a lot of competition check out Frey

**[00:01:50]** ch's video shout out to Frey who

**[00:01:52]** inspired me to build this so we're going

**[00:01:53]** to focus on just the build process so

**[00:01:55]** once you have your topic then you're

**[00:01:58]** obviously going to want to scrape

**[00:01:59]** listings on Google Maps we're focused on

**[00:02:01]** local listings here that are more

**[00:02:03]** Evergreen that people would need in

**[00:02:05]** their like neighborhood so first we're

**[00:02:07]** going to go to Google Maps and we're in

**[00:02:09]** Miami area here it's already searching

**[00:02:11]** for monor schools let's just come up

**[00:02:13]** with something new let's say we're

**[00:02:15]** looking for um shoe repair shops in

**[00:02:18]** Florida let's just say Florida McFarland

**[00:02:21]** shoe repair okay shoe repair okay so we

**[00:02:23]** got some results here we're going to

**[00:02:25]** make a shoe repair website shoe repair

**[00:02:28]** directory all right for in his YouTube

**[00:02:30]** video recommends using a website like

**[00:02:33]** outscar where you can actually it's kind

**[00:02:35]** of cool cuz you can pay with Bitcoin um

**[00:02:37]** you pay PayPal whatever you want but

**[00:02:39]** it's got a Google Maps scraper basically

**[00:02:40]** so you so we would say

**[00:02:42]** Sho uh repair shop great shoe repair

**[00:02:46]** shop and we want to say just Florida

**[00:02:47]** here so we're choosing this Florida all

**[00:02:51]** right unlimited results enhanced results

**[00:02:53]** with other services we don't need any of

**[00:02:55]** that in some cases it might come in

**[00:02:57]** handy but contains one of sh repair shop

**[00:02:59]** delete duplicates use zip codes all

**[00:03:01]** right so we're going to click get data

**[00:03:03]** it's going to tell you how much money

**[00:03:04]** it's going to cost from $9 to

**[00:03:06]** $61 um I'm not going to pay for that but

**[00:03:09]** you can if you want instead I'm going to

**[00:03:10]** show you how to use a script on your own

**[00:03:13]** computer that just uses Google API

**[00:03:15]** credits that you actually get free I

**[00:03:18]** think $300 free credit when you sign up

**[00:03:20]** at the time of me recording this so you

**[00:03:22]** can technically actually get it all for

**[00:03:24]** free yeah so you can see here I've got

**[00:03:26]** $141 of free credit so we're going to

**[00:03:29]** use that to scrape this data okay so

**[00:03:31]** first things first is you're going to

**[00:03:32]** want cursor cursor is an AI coding app

**[00:03:35]** uh it's a lot like Visual Studio code

**[00:03:37]** where basically you're going to have all

**[00:03:38]** these files over here you're going to

**[00:03:40]** have access to your git repository lots

**[00:03:42]** of plugins but there's an AI agent over

**[00:03:45]** here that helps you do anything the

**[00:03:46]** agent will build the ask agent normally

**[00:03:49]** uses a different model and uh it just

**[00:03:51]** helps you figure things out um I have

**[00:03:53]** never really used edit yet but it's

**[00:03:54]** probably for editing software but the

**[00:03:55]** agent helps you build so we're here we

**[00:03:58]** have nothing so I'm going to make a I'm

**[00:04:00]** not even going to do anything yet I'm

**[00:04:01]** going to say write a script that uses

**[00:04:05]** Google Maps

**[00:04:07]** API um I think it's called places um to

**[00:04:12]** scrape Google maps of all listings in

**[00:04:17]** the State of

**[00:04:19]** Florida for the term shoe repair shop

**[00:04:24]** export the results as a CSV file which

**[00:04:29]** includes one rating and then I'm Al okay

**[00:04:32]** so let's just do that okay one how about

**[00:04:34]** we say this one positive rating and one

**[00:04:37]** negative rating so maybe you want more

**[00:04:39]** ratings um the ratings is something I'm

**[00:04:41]** going to integrate a system into monor

**[00:04:43]** find later but for now it's good to

**[00:04:45]** start with ratings and we're just doing

**[00:04:47]** a sample video here so I'm just going to

**[00:04:48]** do one rating so Watch What Happens I

**[00:04:50]** send

**[00:04:52]** that okay so it shows python when I did

**[00:04:56]** mon story find I used JavaScript but

**[00:04:57]** that's okay python ends up doing a lot

**[00:04:59]** of the data scraping later when we want

**[00:05:00]** to generate the descriptions so as you

**[00:05:02]** can see over here it's literally coding

**[00:05:05]** everything I just said I want it uh and

**[00:05:08]** it gets even crazier later when we're

**[00:05:09]** going to use bolt to then plug in the

**[00:05:11]** data we scrape into a website and it

**[00:05:12]** just makes a website so this is all

**[00:05:14]** still the back end we can't use bolt yet

**[00:05:16]** so here it says make a new file

**[00:05:18]** requirements. text and you can either

**[00:05:20]** press this check mark here accept or

**[00:05:22]** just uh Apple y command y or just accept

**[00:05:25]** boom uh scrape sheet repair okay now I

**[00:05:28]** made a python script here it is

**[00:05:30]** this in green shows you what it added

**[00:05:34]** and then later when we edit you'll see

**[00:05:35]** things in red things in green red means

**[00:05:37]** it's cutting it green means it's adding

**[00:05:38]** it so it just wrote this whole script um

**[00:05:41]** now you can obviously vet the script and

**[00:05:43]** look through it and if you really want

**[00:05:45]** to learn how to understand coding then

**[00:05:47]** you simply go in here and Google

**[00:05:50]** everything you don't understand or you

**[00:05:51]** copy it and you say what does that do

**[00:05:55]** but let's do that in a minute so I can I

**[00:05:56]** can just copy it and say like what if I

**[00:05:58]** don't want what if the AI is just

**[00:06:00]** putting malicious code on my computer um

**[00:06:02]** I'm going to add it to the chat and I

**[00:06:03]** can say what does this do and as you can

**[00:06:06]** see here it knows lines 36 to 44 which

**[00:06:08]** is what I highlighted that's its context

**[00:06:11]** and then me explains what it does now I

**[00:06:13]** Ed the agent for that I should have used

**[00:06:14]** ask if anything it'll just save the

**[00:06:16]** credits but whatever all right so then

**[00:06:19]** it tells me exactly what this does it

**[00:06:21]** starts from Florida Center

**[00:06:23]** coordinates latitude blah blah blah and

**[00:06:25]** so I just copied these lines right here

**[00:06:28]** there's obviously a whole script here so

**[00:06:30]** if you just go through the whole thing

**[00:06:31]** you can actually learn what each thing

**[00:06:32]** does you don't need to be a coder

**[00:06:33]** anymore you just have to learn how to

**[00:06:34]** tell it what to do so anyway I'm going

**[00:06:37]** back to up here and it's giving me the

**[00:06:40]** script and I say accept file boom it

**[00:06:43]** accepts it and it'll save it

**[00:06:44]** automatically and then you'll notice

**[00:06:46]** here it's loading environmental

**[00:06:48]** variables we're going to need one of

**[00:06:49]** those files as well because we're going

**[00:06:52]** to need an API key uh and then a read me

**[00:06:54]** so we say yes to the readme okay so then

**[00:06:56]** it gave us everything and it's cool is

**[00:06:58]** in that readme it tells us what to do

**[00:07:00]** it says Okay first install the

**[00:07:01]** requirements then you create a n file

**[00:07:04]** with your Google Maps key so I'm going

**[00:07:06]** to copy that I'm going to make a new

**[00:07:08]** file right here call it. EnV boom now

**[00:07:12]** I'm not going to show you this screen

**[00:07:14]** again but basically I'm going to paste

**[00:07:16]** the Google Maps key right there so we

**[00:07:18]** simply go to Google Cloud I've got my

**[00:07:20]** monor Schools project here you're going

**[00:07:21]** to have to make a new project you can

**[00:07:22]** Google it it's fairly straightforward go

**[00:07:24]** to apis and services go to credentials

**[00:07:27]** and you're going to create credentials

**[00:07:29]** you're going to create an API key and

**[00:07:32]** then it's going to give it to you and

**[00:07:33]** you're going to paste it into your end

**[00:07:35]** file now if you've never coded before

**[00:07:37]** you might have to install python on your

**[00:07:39]** computer or JavaScript or whatever

**[00:07:41]** cursor asks you to do in this case I

**[00:07:43]** already have Pip installed uh if we look

**[00:07:45]** at a command line pip means like what

**[00:07:48]** software we're using pip or python or

**[00:07:49]** node npm that's like the general like

**[00:07:52]** what language does it need to use to

**[00:07:55]** launch the script and then this is the

**[00:07:56]** command install and then boom it's it's

**[00:07:59]** going to install whatever it reads from

**[00:08:01]** the requirements file so we're going to

**[00:08:02]** open a terminal new terminal we want to

**[00:08:04]** run pip install requirements now we're

**[00:08:07]** going to have an issue here maybe

**[00:08:09]** because on my computer like we'll

**[00:08:10]** definitely have an issue here with

**[00:08:11]** python script Sho repair we're going to

**[00:08:12]** need to use Python 3 Let's see looks

**[00:08:15]** like it worked okay so it's installing

**[00:08:17]** all that if you get an error you copy it

**[00:08:20]** and then you either send it to the chat

**[00:08:22]** or you Google it and then it'll tell you

**[00:08:24]** what to do this is a normal part of

**[00:08:26]** coding and processing something will

**[00:08:28]** always go wrong and we need to be

**[00:08:30]** confident and Sovereign enough in our

**[00:08:32]** abilities to figure out the issue it's

**[00:08:34]** easier than ever now thanks to llms so

**[00:08:38]** I'm making this I'm making this video

**[00:08:40]** now cuz I'm confident doing it and I

**[00:08:41]** think you can be confident too just

**[00:08:43]** takes experience we only get experience

**[00:08:46]** by building okay now we have an

**[00:08:48]** error this is a lot so see how it even

**[00:08:50]** gives me this button add to chat see how

**[00:08:53]** it's got the context here the lines and

**[00:08:55]** I just say what happened not fully

**[00:08:58]** compatible let me modify the

**[00:09:00]** requirements so basically it knows I'm

**[00:09:01]** using python 313 but it's not compatible

**[00:09:05]** with that version of pandas which it

**[00:09:06]** needs so it's changing the requirement

**[00:09:08]** the dependency here I accept it I save

**[00:09:10]** it and then I simply press up uh on the

**[00:09:14]** terminal and it's going to go back to

**[00:09:15]** that pip install requirements and I

**[00:09:18]** press enter to run it let's give that a

**[00:09:20]** minute it looks like it happened fast

**[00:09:22]** boom there we go we got it now the next

**[00:09:25]** thing we got to do if we scroll back up

**[00:09:30]** we can run the script so if I try to run

**[00:09:32]** it now it's probably going to give me an

**[00:09:33]** error because it doesn't have python but

**[00:09:34]** it does have Python 3 so it's scrape

**[00:09:38]** shoe repair. python okay now let's do

**[00:09:43]** it now see how it's got some it's called

**[00:09:46]** Uh I don't even know it's called verbose

**[00:09:47]** logs in there so basically it's kind of

**[00:09:49]** telling you what it's doing data

**[00:09:51]** exported okay it found 24 shops great

**[00:09:53]** that probably wasn't very expensive

**[00:09:55]** hopefully it wasn't very expensive so it

**[00:09:58]** gave us something we haven't looked at

**[00:10:00]** it yet and we don't know if it did it

**[00:10:01]** right so this is what's really important

**[00:10:04]** as as we get AIS to do things for us

**[00:10:06]** especially if we don't know how to do

**[00:10:07]** them ourselves we don't know if they're

**[00:10:09]** going to work right but that's why we do

**[00:10:11]** step- by-step in iterations so let's

**[00:10:13]** take a look at the CSV file looks like

**[00:10:15]** it made all these columns in fact let's

**[00:10:17]** just open this in a CSV Viewer okay so I

**[00:10:21]** hope this isn't too small we've got name

**[00:10:23]** address rating total ratings latitude

**[00:10:25]** longitude positive review negative

**[00:10:27]** review if they got one the CSV looks

**[00:10:29]** good but I'm realizing we forgot a

**[00:10:30]** couple important things mainly the

**[00:10:32]** website so I'm going to exit that out

**[00:10:34]** and I'm going to say add a column for

**[00:10:37]** website and description um into the into

**[00:10:42]** the scrape shoe repair

**[00:10:45]** script okay so we're it gave us his

**[00:10:47]** changes let's add those let's say yes

**[00:10:50]** cool accept and then say we also need to

**[00:10:53]** add a zip code

**[00:10:57]** separate full address street address ZIP

**[00:11:01]** code and city into their own unique

**[00:11:06]** columns okay so even if we realize we

**[00:11:09]** need more data here this is just what

**[00:11:11]** it's going to be because this is just a

**[00:11:12]** tutorial video but as you can see we can

**[00:11:14]** keep enhancing this script in the

**[00:11:16]** beginning we obviously don't want to use

**[00:11:19]** too much data to for our apri credits to

**[00:11:21]** pay money until we know exactly what we

**[00:11:23]** want I accept that it's got street

**[00:11:25]** address city state ZIP code full address

**[00:11:27]** okay great um okay so it's doing

**[00:11:30]** everything we want now can we also

**[00:11:32]** scrape for a description of the

**[00:11:35]** listing now we are going to use an llm

**[00:11:38]** to scrape the listings website but this

**[00:11:41]** is also a good move just in case like

**[00:11:44]** Florida Sho repair how many websites do

**[00:11:46]** they have I mean maybe they have lots of

**[00:11:48]** websites or maybe they're these old mom

**[00:11:49]** and pop businesses that just run off of

**[00:11:51]** board and mouth and they don't think

**[00:11:52]** they need a website and they don't cuz

**[00:11:53]** their business is still going um we're

**[00:11:55]** going to try both ways here so we're

**[00:11:56]** going to accept the accept there okay

**[00:11:59]** okay so now we're going to run this

**[00:12:01]** script again all I got to do is press up

**[00:12:04]** python scrape shoe repair I'm going to

**[00:12:05]** open up our original CSV and just delete

**[00:12:07]** it just so there's no potential issues

**[00:12:09]** there Python 3 scrapes shoe repair. piy

**[00:12:12]** boom okay it worked so now after few

**[00:12:15]** back and forths we're getting the

**[00:12:18]** address the full address the country the

**[00:12:20]** state the city the ZIP code we got the

**[00:12:22]** phone number we got the URL and we got a

**[00:12:26]** negative review and a positive review

**[00:12:28]** great so now we have to work with and we

**[00:12:30]** got 42 listings that's like perfect

**[00:12:32]** actually it's less than that I think

**[00:12:33]** it's 24 24 listings great little mini

**[00:12:36]** tutorial website so the next thing I'm

**[00:12:37]** going to do uh I'm going to clear my

**[00:12:40]** files out here a bit I'm going to make a

**[00:12:41]** new file and I'm going to call it

**[00:12:43]** organized data. piy and I'm going say

**[00:12:48]** in this file and I'm going to tag it

**[00:12:52]** organized data. py write a script that

**[00:12:55]** removes listings from our CSV file that

**[00:12:59]** has no

**[00:13:01]** address or no

**[00:13:03]** state if there is no

**[00:13:07]** website um add no

**[00:13:10]** website uh but where are we going to add

**[00:13:12]** it so let's do this um add no website

**[00:13:17]** yeah to the website column let's just

**[00:13:19]** add that there for now so what that's

**[00:13:21]** going to do is just parse everything and

**[00:13:23]** say hey this one doesn't have an address

**[00:13:24]** this one might not be real so just

**[00:13:26]** remove it and I'm going to rename our

**[00:13:28]** CSV file to just data. CSV so it's

**[00:13:32]** easier all right so now it's writing an

**[00:13:34]** organization script to just filter out

**[00:13:36]** if anything's broken uh right here it

**[00:13:38]** says hey you know choose your CSV file

**[00:13:41]** what I like to do so we can use this

**[00:13:43]** again later is send it to the chat and

**[00:13:46]** say make it so I can choose the

**[00:13:50]** CSV file in the command line and the

**[00:13:53]** output file is simply file name updated.

**[00:13:58]** CSV now we got organized data and it

**[00:14:00]** says to run it all we do is say

**[00:14:03]** organized data your your input file to

**[00:14:05]** CSV so I say Python 3 for you it might

**[00:14:08]** be python for me it's Python 3 we're

**[00:14:11]** going to run organize data. piy and

**[00:14:13]** we're going to data. CSV boom so now we

**[00:14:16]** got data updated over here and

**[00:14:21]** um let's do this again let's add a

**[00:14:24]** success message that

**[00:14:27]** shares how many

**[00:14:29]** um rows were updated or deleted so we

**[00:14:33]** don't even know what happened so let's

**[00:14:36]** add in that

**[00:14:42]** message and what you can do also the

**[00:14:44]** next level if you have a lot of listings

**[00:14:46]** it say send all the failed ones to a new

**[00:14:49]** CSV file so I can just check them

**[00:14:51]** manually and that is a step that I'm

**[00:14:53]** still working on and organizing my data

**[00:14:55]** from monor find okay so it added this

**[00:14:57]** new stuff let's run it

**[00:14:59]** again now it says updated entries four

**[00:15:03]** so to four entries it added no website

**[00:15:06]** perfect so we're going to go to Swift

**[00:15:08]** Shoe Repair Inc because it said no

**[00:15:16]** website okay there's

**[00:15:18]** Swifts um and it's true doesn't look

**[00:15:20]** like it has a website um bag clinic and

**[00:15:23]** shoe

**[00:15:24]** repair does look real and open this is

**[00:15:26]** how we enhance the data you know we just

**[00:15:28]** check it out 5 years ago when were these

**[00:15:30]** reviews 3 months ago okay so it's still

**[00:15:33]** there um but then I would just go

**[00:15:34]** through each one of these ones with no

**[00:15:36]** URL and um and just check to see if

**[00:15:40]** they're still around boom um then we see

**[00:15:43]** missing address is none address without

**[00:15:45]** Florida none okay so there we have our

**[00:15:47]** file it's clean depending on your data

**[00:15:49]** you might have a lot to do here and it

**[00:15:51]** might take a while um or you do it in a

**[00:15:53]** future round but in general I try to at

**[00:15:56]** least get out like the dead ones before

**[00:15:58]** we even go live so now we have our data

**[00:16:00]** here okay data updated. CSV and now

**[00:16:04]** we're basically ready to start a brand

**[00:16:06]** new project here okay so we've got our

**[00:16:08]** base data but now we need to make a

**[00:16:09]** description in order to do that we need

**[00:16:12]** to scrape the URLs of each website in

**[00:16:14]** here go through it find the important

**[00:16:17]** information and then generate a

**[00:16:18]** description from it so for that we're

**[00:16:20]** going to use an llm AI and we're going

**[00:16:23]** to use software called crawl for AI

**[00:16:25]** which is an open source llm friendly web

**[00:16:27]** crawler and scraper and and it looks a

**[00:16:29]** little intimidating it was for me in the

**[00:16:31]** beginning but it actually is going to

**[00:16:32]** make a lot of sense as we go through it

**[00:16:34]** so first things first we are going to

**[00:16:38]** make a fresh file and we'll just call it

**[00:16:41]** description

**[00:16:42]** generator. Pi okay now the first thing

**[00:16:46]** we need to do if we just make a quick

**[00:16:47]** list we need to connect to the LM AI

**[00:16:51]** which is going to be Venice and then we

**[00:16:53]** got to uh so see and cursor it gets you

**[00:16:55]** all this stuff from the GetGo um but it

**[00:16:58]** doesn't know that we're not really

**[00:16:59]** programming here then we need

**[00:17:03]** to uh scrape the website for two to

**[00:17:08]** three paragraphs of valid information

**[00:17:11]** valid text and then we're going to

**[00:17:14]** generate a description of the website

**[00:17:15]** and we're going to Output the

**[00:17:17]** description to the column uh to the

**[00:17:21]** description column of the CSV file we're

**[00:17:26]** going to repeat the process save the

**[00:17:28]** updated yeah there we go um okay so

**[00:17:32]** that's what we're going to do here and

**[00:17:33]** we're actually going to copy and paste

**[00:17:34]** that into the agent in a minute but

**[00:17:36]** let's just first say okay how are we

**[00:17:38]** going to scrape the website well we're

**[00:17:40]** going to use crawl for AI and we're

**[00:17:42]** going to use um what's basically called

**[00:17:45]** llm extraction method which is llm

**[00:17:47]** strategies here why would you use an llm

**[00:17:50]** for scraping well it's going to help

**[00:17:51]** structure the data if it's unstructured

**[00:17:53]** the website itself it can reason through

**[00:17:55]** it and be like okay that's necessary

**[00:17:56]** that's not necessary just in general why

**[00:17:58]** do we use l l m because they help us do

**[00:18:00]** tedious things faster and this is very

**[00:18:02]** definitely a tedious thing so the first

**[00:18:04]** thing I'm going to do I've opened the

**[00:18:06]** agent here and we're going to start

**[00:18:08]** telling it what we want to do here but

**[00:18:10]** the reason I made these steps is that it

**[00:18:12]** can be a little complicated here so the

**[00:18:14]** reason I made these steps is because

**[00:18:16]** this can be complicated to tell the

**[00:18:18]** agent to do it all at once so we're

**[00:18:20]** basically going to test it step by step

**[00:18:22]** until we know each step is working so

**[00:18:24]** then by the time we run it we know it

**[00:18:26]** can do what we want we could just say

**[00:18:27]** hey

**[00:18:29]** do this and it just sends it all of

**[00:18:32]** there we could just say hey do this and

**[00:18:35]** paste that all in

**[00:18:37]** there but the truth is probably not

**[00:18:40]** going to work very well so we're going

**[00:18:42]** to start from the top first thing we're

**[00:18:44]** going to do here is go is connect to the

**[00:18:48]** llm AI so I'm going to actually make a

**[00:18:50]** new document just going to call it

**[00:18:52]** steps. MD or even steps. text we have

**[00:18:55]** the steps we want to do so we are going

**[00:18:58]** to write a script that's tell it what

**[00:19:01]** it's called it's going to be called um

**[00:19:03]** description generator. piy that scrapes

**[00:19:06]** URLs in our data updated. CSV file for

**[00:19:13]** information to summarize into a brief

**[00:19:17]** summary first let's connect to the

**[00:19:20]** Venice AI API now we can't just say that

**[00:19:24]** with open AI maybe anthropic yeah but

**[00:19:26]** Venice isn't as deep into the system so

**[00:19:29]** we're going to go to Venice ai's

**[00:19:31]** documentation and in here we're going to

**[00:19:33]** go to their reference and basically I

**[00:19:36]** don't know what a Swagger definition is

**[00:19:37]** but in fact this is actually perfect

**[00:19:39]** we're going to take that Swagger that

**[00:19:42]** yaml Venice actually let's make a folder

**[00:19:45]** called Venice

**[00:19:47]** docs okay and in Venice docs we're going

**[00:19:50]** to make a file called Swagger DOL going

**[00:19:53]** to paste all that in there and that has

**[00:19:55]** like all these little things that the

**[00:19:58]** script is going to need to know in order

**[00:19:59]** to communicate with the Venice aai

**[00:20:01]** through their API including the URL that

**[00:20:03]** might be enough um but basically I'm

**[00:20:07]** going to copy all this in there too and

**[00:20:09]** I'm going to say what do we call

**[00:20:12]** Basics Mt copy that in there um all

**[00:20:16]** right so now we've copied and pasted

**[00:20:19]** whenever you're going to use an API or a

**[00:20:20]** script which we're going to use with

**[00:20:21]** craw for AI copy and paste it into your

**[00:20:24]** workspace in a file so that way you can

**[00:20:27]** drag this in and say okay write a script

**[00:20:30]** we're going to write a script but first

**[00:20:32]** let's connect to the Venice AI API

**[00:20:34]** successfully check Venice

**[00:20:37]** docks and add it to the python

**[00:20:41]** script all right so now I hadn't saved

**[00:20:45]** that but hopefully it's

**[00:20:52]** okay so I've already got the Venice API

**[00:20:55]** key in there and that's giv me the code

**[00:20:58]** and I'm going to apply that to

**[00:20:59]** description generator except file now

**[00:21:01]** you can see it opens open AI cuz it uses

**[00:21:03]** open ai's uh module and you just change

**[00:21:06]** a few things to Venice looks like it's

**[00:21:08]** got the right base URL um I've been

**[00:21:11]** working with Venice AI enough to know

**[00:21:13]** that this is working uh so okay we're

**[00:21:15]** going to save that and now we are going

**[00:21:18]** to install these

**[00:21:23]** dependencies okay they're all installed

**[00:21:25]** now now we're going to do Python 3

**[00:21:28]** description generator.

**[00:21:31]** py um it failed to do everything but it

**[00:21:34]** looks like we didn't it yeah so add a

**[00:21:38]** command so we can just test the API

**[00:21:43]** connection all right so now it's just

**[00:21:47]** going to add this little this um this

**[00:21:50]** text basically so once we connect it

**[00:21:52]** says hey it's work and we just have to

**[00:21:55]** run Python 3

**[00:21:59]** description generator test so this flag

**[00:22:01]** here is just saying hey we're going to

**[00:22:02]** test it that you can add like okay it it

**[00:22:05]** failed so we're going to send this what

**[00:22:11]** happened and our lovely AI agent is

**[00:22:14]** going to tell us what

**[00:22:17]** happened and let's

**[00:22:20]** see okay so it's going to change

**[00:22:22]** whatever it needs to change save it

**[00:22:24]** let's try that

**[00:22:25]** again testing connection successful all

**[00:22:28]** right so that's step one all right now

**[00:22:31]** let's go back to our steps and we got

**[00:22:34]** okay so that one's done two scrape the

**[00:22:36]** website all right so here's where crawl

**[00:22:39]** for AI comes in so crawl for AI here is

**[00:22:42]** a really awesome script here and

**[00:22:45]** basically as we're going through we can

**[00:22:47]** obviously ask the AI to write a code for

**[00:22:48]** us but code creation is not simple

**[00:22:51]** there's a reason it's a high paying

**[00:22:53]** career F you know it's it's a big deal

**[00:22:55]** it's been a big deal for so long cuz it

**[00:22:57]** just takes time things don't always work

**[00:22:58]** right even if you're in AI so while it

**[00:23:00]** is getting better it still is great to

**[00:23:02]** use project repositories Frameworks out

**[00:23:05]** there that already built for you and

**[00:23:06]** then the AI doesn't have to build it it

**[00:23:08]** just says oh use that use that use that

**[00:23:10]** so what we're going to do is we're going

**[00:23:11]** to install we're going to do the quick

**[00:23:13]** start here and we're going to install

**[00:23:15]** and then we're going to implement crawl

**[00:23:17]** for AI into our project that's

**[00:23:20]** installed

**[00:23:23]** setup

**[00:23:25]** boom and then now it says verify your

**[00:23:27]** installation make sure sure it's working

**[00:23:29]** that's another python script going to

**[00:23:31]** check through example.com it's going to

**[00:23:33]** pull 300 characters so I'm just going to

**[00:23:35]** copy that I'm going to make a new

**[00:23:36]** document called test. python put that in

**[00:23:40]** there save it and then just say Python 3

**[00:23:43]** test. python it's going to test this

**[00:23:44]** connection to example.com and it worked

**[00:23:47]** got example domain this domain is that

**[00:23:49]** that's what it that's what it scraped

**[00:23:51]** all right so craw fre ey is open so back

**[00:23:53]** to our steps so we want to scrape the

**[00:23:55]** website and generate a description so

**[00:23:57]** first off scrape the website so next we

**[00:24:00]** want to scrape the U the

**[00:24:03]** website URL from the from the listing

**[00:24:07]** Row in data updated. CSV and we're just

**[00:24:11]** going to say find two to three

**[00:24:14]** paragraphs of relevant information to

**[00:24:18]** summarize the

**[00:24:20]** listing we can even say summarize the

**[00:24:22]** shoe repair store format it to markdown

**[00:24:26]** let's just say prune filter it and then

**[00:24:30]** format it to markdown so that's the

**[00:24:32]** first thing we're going to do um

**[00:24:33]** basically and I I know that process

**[00:24:36]** because I've been using it and I'm going

**[00:24:37]** to walk you through it here now I

**[00:24:39]** recommend you go through all of this

**[00:24:40]** especially if you're going to do a lot

**[00:24:41]** of data scraping but basically you copy

**[00:24:44]** and paste what you want from the read me

**[00:24:48]** just like we did with the Venice API

**[00:24:49]** specs copied it into a document you do

**[00:24:52]** the same thing here for this script so

**[00:24:54]** then you you can just say hey use

**[00:24:56]** browser config module use crawler run

**[00:24:58]** config module or if you're doing deep

**[00:25:00]** crawling use uh use uh deep crawl

**[00:25:04]** strategy llm web scraping strategy so

**[00:25:07]** instead of just blindly telling the AI

**[00:25:10]** to do it you give it the documents so

**[00:25:12]** you go to GitHub you just download the

**[00:25:14]** zip and then I copied the docs folder

**[00:25:17]** into cursor here and then here in mdv2

**[00:25:21]** that's really where we see everything so

**[00:25:23]** I'm going to document I'm going to drag

**[00:25:25]** mdv2 which are the docs and I'm going to

**[00:25:27]** put that

**[00:25:29]** uh what needs to be higher up Okay C

**[00:25:33]** documentation in

**[00:25:35]** mdv2 now one thing we need to know

**[00:25:38]** though is what modules do we want to use

**[00:25:40]** use

**[00:25:41]** modules all right so first off we're

**[00:25:45]** going to that's simple crawling it's

**[00:25:46]** basically going to do that on its own

**[00:25:48]** we're definitely going to use async web

**[00:25:49]** crawler because that's the very basics

**[00:25:51]** of

**[00:25:52]** everything um async web crawler we going

**[00:25:56]** to need crawler result I believe

**[00:25:59]** um yeah cuz this results

**[00:26:03]** everything it puts out everything it

**[00:26:05]** finds um llm config um we're going to

**[00:26:09]** use LM

**[00:26:13]** config we're going to use markdown

**[00:26:16]** generation to clean it

**[00:26:21]** up uh we're going to use fit markdown

**[00:26:24]** and in order which is basically using a

**[00:26:26]** Content filter like we mentioned so

**[00:26:27]** we're going to use pruning content

**[00:26:29]** filter here once again I'm very familiar

**[00:26:31]** with this but you might have different

**[00:26:32]** purposes like bm25 could be good if you

**[00:26:36]** want to scrape search results for

**[00:26:37]** example so make sure you go through all

**[00:26:40]** this for your own use case I'm just

**[00:26:42]** doing kind of the the be Basics here we

**[00:26:45]** do want to make a cach so cach

**[00:26:48]** mode that'll kind of speed things up a

**[00:26:51]** little bit all right so I think that's

**[00:26:53]** good we'll just say modules to consider

**[00:26:59]** all right so it's not going to generate

**[00:27:00]** it yet it's not going to generate

**[00:27:02]** anything yet it's just going to scrape

**[00:27:04]** the website and get the two to three

**[00:27:05]** paragraphs of text how about we say

**[00:27:07]** format it to markdown create a unique

**[00:27:11]** directory per listing with its own

**[00:27:14]** markdown file so it has the raw data we

**[00:27:17]** got the two to three paragraphs okay so

**[00:27:19]** we're going to send that out now and

**[00:27:20]** we're going to see how it

**[00:27:22]** does and once again it has the

**[00:27:24]** documentation here for the script

**[00:27:26]** doesn't mean it's going to do it right

**[00:27:27]** but it looks so so far like it's doing

**[00:27:29]** it right and if we said connect to vsai

**[00:27:32]** and now scrape the website and now

**[00:27:33]** generated description all at once it

**[00:27:35]** wouldn't work in fact I recorded this

**[00:27:37]** for an hour before and I did that and it

**[00:27:38]** didn't work so I'm recording it again um

**[00:27:40]** so that's a very important lesson I'm

**[00:27:42]** trying to pass on to you there are the

**[00:27:44]** changes here you're seeing it's

**[00:27:46]** importing more modules it's added all

**[00:27:47]** this code okay accept file okay so let's

**[00:27:50]** see what happens here we got pyone 3 the

**[00:27:53]** last thing we want to see is uh where is

**[00:27:55]** it getting yeah it's getting the right

**[00:27:57]** CSV file

**[00:27:58]** Python

**[00:28:00]** 3 description generator. py okay so we

**[00:28:03]** got an error there um we highlight that

**[00:28:06]** we add it to the chat what

**[00:28:09]** happened uh we switched to ask for that

**[00:28:13]** basically ask I believe you use like a

**[00:28:15]** cheaper llm and then agent you use a

**[00:28:17]** more powerful more expensive llm smarter

**[00:28:20]** one all right so description generator

**[00:28:23]** all right so whatever I don't know what

**[00:28:24]** it did um oh you have to do it somewhere

**[00:28:26]** else

**[00:28:28]** whatever that is okay let's

**[00:28:30]** see still not working let's send it to

**[00:28:33]** the chat what

**[00:28:37]** happened oh default markdown generat in

**[00:28:39]** a different module so yeah I mean the AI

**[00:28:42]** has to figure stuff out just like us

**[00:28:44]** humans so the simpler you make and the

**[00:28:46]** more concise all the directions the more

**[00:28:49]** chance it'll be done accurately the

**[00:28:51]** first time but of course you can okay so

**[00:28:53]** the the script ran but of course you can

**[00:28:54]** always just do it like this and just

**[00:28:56]** debug as you go um all right so we got

**[00:28:59]** an error there let's send it

**[00:29:01]** here

**[00:29:04]** happened prob URL right it didn't have

**[00:29:07]** anything to

**[00:29:09]** crawl still didn't work we're going to

**[00:29:12]** say error again are we sure it's pulling

**[00:29:16]** the URL from the

**[00:29:18]** data updated. CSV file he so this part

**[00:29:22]** of the video is kind of boring cuz I go

**[00:29:23]** back and forth a bit so I'm going to cut

**[00:29:25]** back in So once we solve the problem of

**[00:29:26]** it finding the URL we go on to the next

**[00:29:28]** problem the key Point here is solve one

**[00:29:31]** problem at a time don't try to solve

**[00:29:33]** multiple problems with one prompt it

**[00:29:35]** will just make more problems probably

**[00:29:37]** okay wait all right it's crawling it's

**[00:29:38]** crawling okay we still failed but we got

**[00:29:41]** some progress crawl result as no okay

**[00:29:45]** so uh oh let's see we see that at least

**[00:29:50]** it is scraping data it's getting the

**[00:29:52]** original stuff yeah so there we go

**[00:29:54]** that's probably like part of the

**[00:29:55]** description quality so it's doing the

**[00:29:56]** scraping right that much is working but

**[00:29:59]** what's not working is whatever is next

**[00:30:02]** so yeah generating a description

**[00:30:08]** so not working it's something to do with

**[00:30:11]** the Venice

**[00:30:13]** API we need to remove Venice parameter

**[00:30:15]** since we're using the async open AI

**[00:30:17]** client fix the generate okay yeah so

**[00:30:20]** basically this is just slight things to

**[00:30:22]** do I think between open AI

**[00:30:24]** versus Venice AI apis

**[00:30:28]** let's see let's

**[00:30:30]** see oh I think it

**[00:30:34]** worked processing Row one all right so

**[00:30:37]** let's cancel that and let's just see we

**[00:30:40]** should have updated file

**[00:30:43]** here it generated a description we just

**[00:30:46]** didn't tell it where to put it and

**[00:30:48]** that's perfect because all we wanted to

**[00:30:50]** do is scrape website now it is

**[00:30:52]** generating the description so we've got

**[00:30:54]** the description so now we need to say um

**[00:30:57]** make make sure the

**[00:30:59]** script make sure the script outputs the

**[00:31:02]** description the generated description to

**[00:31:05]** the description

**[00:31:06]** column of the uh of an updated CSV file

**[00:31:12]** call that data

**[00:31:15]** updated uh descriptions that's CSP and

**[00:31:18]** we're almost

**[00:31:20]** done I think it's already doing

**[00:31:25]** this these last two might already be

**[00:31:27]** built in we're going to find out okay so

**[00:31:30]** there's our code we'll accept everything

**[00:31:33]** there and I've got a good feeling that

**[00:31:35]** we're we're getting

**[00:31:37]** close

**[00:31:38]** boom so it saved the markdown file not

**[00:31:42]** seeing a new CSV file I have a feeling

**[00:31:45]** it's doing it in the end which is why we

**[00:31:49]** wrote these

**[00:31:51]** steps so it's probably just putting the

**[00:31:53]** descriptions it's generating into its

**[00:31:55]** memory which might be adding extra

**[00:31:58]** context yeah and then it makes the file

**[00:31:59]** at the end exactly what it did

**[00:32:02]** here

**[00:32:03]** so there's a brief

**[00:32:07]** description all right so let's open this

**[00:32:10]** it's back here where is it right here

**[00:32:12]** descriptions all right now we obviously

**[00:32:14]** need to work on this prompt here's a

**[00:32:15]** brief description of the shoe store um

**[00:32:19]** yeah it's not working but it's close

**[00:32:22]** Okay so at least it's like doing what we

**[00:32:24]** want it's just not doing it right it

**[00:32:26]** skipped the no website on that's good

**[00:32:29]** they did have this here so that's the

**[00:32:31]** description I gave first Sho repair also

**[00:32:33]** known as America's gobler offers expert

**[00:32:36]** shoe repair okay I mean they do it too

**[00:32:38]** here so all right so it's good but it's

**[00:32:41]** no cigar so we're going to

**[00:32:46]** say um we don't even need to say this to

**[00:32:49]** the agent let's just go to the

**[00:32:52]** prompt and let's find

**[00:33:00]** you are a helpful assistant that creates

**[00:33:02]** concise informative descriptions of sh

**[00:33:04]** repair businesses so this is the prompt

**[00:33:05]** it's giving the llm focus on key

**[00:33:08]** Services Specialties and unique

**[00:33:09]** features um okay and then we just say um

**[00:33:14]** do not output anything else with the

**[00:33:17]** description thank you autofill um and we

**[00:33:19]** all say do not make anything up if

**[00:33:23]** there's not enough data say not enough

**[00:33:26]** data okay so now we change that prompt

**[00:33:30]** and um let's just say create a two to

**[00:33:33]** three sentence

**[00:33:35]** description all right so now I'm going

**[00:33:37]** to delete that new CSV file so we don't

**[00:33:40]** get all messed up and we're going to run

**[00:33:41]** this new script

**[00:33:43]** again if this is happening how it should

**[00:33:46]** be which I'm not sure it is but it's got

**[00:33:48]** these content MD files already there

**[00:33:51]** that's just generating the descriptions

**[00:33:52]** from there's one other thing we need to

**[00:33:54]** do still which is actually steps five

**[00:33:56]** and six together which is well it's

**[00:33:59]** already repeating the process but then

**[00:34:00]** save the updated CSV file as it goes so

**[00:34:04]** let's save the rows with descriptions to

**[00:34:09]** the new CSV file

**[00:34:13]** one by one instead of

**[00:34:18]** waiting so we're just going to have to

**[00:34:19]** do that and then we're done with the

**[00:34:20]** steps

**[00:34:22]** document um once again we go to the

**[00:34:24]** descriptions

**[00:34:26]** here and let's just see how that line

**[00:34:29]** did there we go McFarland shoe repair

**[00:34:31]** also known as am's cobbler offers high

**[00:34:32]** quality shoe repair services Drive shoe

**[00:34:34]** repair not enough data Phenix muled

**[00:34:36]** shoes inserts and braces Redwing shoe

**[00:34:38]** store in Lakeland Florida okay so we got

**[00:34:40]** it okay the last thing we would do like

**[00:34:42]** I said we had say hey write it one by

**[00:34:44]** one instead of all the time at the end

**[00:34:47]** so if uh we delete that right

**[00:34:49]** now and do that

**[00:34:51]** again we're going to notice it makes a

**[00:34:53]** new CSV file

**[00:34:56]** sooner boom so it's there then it's got

**[00:35:00]** uh yeah same description pretty much and

**[00:35:02]** it's just going to add our new ones

**[00:35:04]** there one at a time perfect so we have

**[00:35:07]** some base data now but we're going to

**[00:35:09]** want to clean it a little more which is

**[00:35:11]** why like on this one not enough data

**[00:35:14]** let's go to this website let's see I

**[00:35:16]** mean yeah I see why there's not enough

**[00:35:18]** data

**[00:35:19]** huh um but it's it's there right it's

**[00:35:22]** not actually a shoe store they fix

**[00:35:26]** shoes they kind of fix shoes but we're

**[00:35:29]** looking for

**[00:35:30]** people I'm just going to delete that

**[00:35:33]** that listing uh not enough data let's

**[00:35:35]** check this one there's a Facebook link

**[00:35:38]** that would be why now maybe in another

**[00:35:40]** video we'll go over how we can give it

**[00:35:42]** our Facebook login and like hey log in

**[00:35:44]** and get this information I'm not going

**[00:35:46]** to do that in this video um all right so

**[00:35:48]** we're running it one last time we're

**[00:35:50]** going to have our final data we're going

**[00:35:52]** to delete a few listings and next up

**[00:35:54]** we're going to put it into a website all

**[00:35:57]** right so it's making the new

**[00:35:59]** folder made the new CSV file boom we got

**[00:36:03]** it and it's got descriptions okay great

**[00:36:05]** so we have our data updated

**[00:36:07]** descriptions. CSV okay so we got this

**[00:36:10]** there's one last thing we want to do and

**[00:36:12]** that is take the address and split it

**[00:36:14]** into a few different columns CU we want

**[00:36:16]** to have the city in its own column the

**[00:36:19]** state not so much cuz it's all only just

**[00:36:20]** Florida and the zip code so let's say

**[00:36:23]** let's write let's make a new file and

**[00:36:25]** let's call it um blit

**[00:36:30]** address and we're going to say write a

**[00:36:33]** script as split address that separates

**[00:36:38]** the street address from the

**[00:36:43]** city Z code State and Country into new

**[00:36:50]** columns all right so here we go split

**[00:36:52]** address.

**[00:36:53]** piy um let's see what CSV file is it

**[00:36:59]** using data updated descriptions that's

**[00:37:02]** CSV Alpha file let say data updated

**[00:37:04]** addresses okay fine so this is where

**[00:37:06]** your taxonomy it's getting important

**[00:37:09]** you're going to get all these data files

**[00:37:10]** you got to stay organized I just keep

**[00:37:11]** making new folders after a new uh leap

**[00:37:14]** okay so we're going to run that now

**[00:37:18]** well run

**[00:37:20]** command all right cool so it worked

**[00:37:23]** let's see now let's see we got

**[00:37:29]** address street address city state ZIP

**[00:37:32]** code

**[00:37:33]** country go all the way to the end D

**[00:37:37]** address city state ZIP code country

**[00:37:39]** perfect so now we have all the data we

**[00:37:41]** need now what we're going to do is

**[00:37:43]** convert it to Json convert to json. Pi

**[00:37:47]** and we're going to say we could just put

**[00:37:49]** it here say write a script that converts

**[00:37:55]** data updated addresses that's our latest

**[00:37:58]** one to Json for use in populating a

**[00:38:02]** directory website to a Json

**[00:38:09]** file so CSV is great for spreadsheets uh

**[00:38:13]** it's good for a lot of things it could

**[00:38:14]** also work on a website but Json is the

**[00:38:16]** standard so here we go we're going to

**[00:38:19]** run this script now we're going to

**[00:38:21]** convert our lovely

**[00:38:24]** uh going to convert our lovely CSV file

**[00:38:29]** Python 3 convert json. piy boom and now

**[00:38:33]** you'll see what it looks like

**[00:38:36]** is is stores. Json oh shoe stores great

**[00:38:40]** so now we got stores. Json and it's

**[00:38:42]** organized all that information this is

**[00:38:44]** what a Json file looks like each new

**[00:38:46]** squiggly line means it's a new entry

**[00:38:48]** boom latitude longitude great we got all

**[00:38:51]** the info we need now make our shoe

**[00:38:53]** repair shop in Florida hey guys before

**[00:38:56]** we continue with the best part of this

**[00:38:58]** project making the website I just wanted

**[00:39:00]** to let you know that this is a brand new

**[00:39:02]** channel and if you're still watching if

**[00:39:03]** you're finding value in this I would

**[00:39:05]** absolutely love it if you could press

**[00:39:07]** like and maybe even subscribe if you

**[00:39:09]** want to continue building with me I'm

**[00:39:11]** going to be making videos like this a

**[00:39:12]** lot I love building I love using AI so

**[00:39:15]** if you could do that that would be

**[00:39:17]** really awesome okay let's build this

**[00:39:19]** website we're going to go to bolt. new

**[00:39:22]** there's also bolt. DIY which is a fork

**[00:39:25]** of bolt. new uh and you can run it

**[00:39:28]** locally with local models but it is

**[00:39:30]** pretty slow so um I I'm using bt. new um

**[00:39:35]** I love open source stuff but I tried

**[00:39:36]** both on DIY it just took forever so I

**[00:39:38]** have two two two million monthly tokens

**[00:39:40]** remaining I'm going to try to use less

**[00:39:42]** than half of that I used 8 million for

**[00:39:43]** building monori fine uh and I was

**[00:39:46]** learning bolt. new as I went so I

**[00:39:48]** realized I could have done it for

**[00:39:49]** cheaper I'm going to say build a direct

**[00:39:53]** I'm not even gonna okay you could use

**[00:39:56]** versel nextjs you could study up on

**[00:39:58]** these Frameworks spelt um and choose

**[00:40:01]** which one you want but we're just going

**[00:40:02]** to let it do it okay um it's going to

**[00:40:04]** use nextjs probably um but I'm going to

**[00:40:08]** show you two ways to do this first we're

**[00:40:10]** just going to give it to bolt. New build

**[00:40:12]** a directory uh for shoe repair shops in

**[00:40:18]** Florida uh the listings are in the

**[00:40:22]** attached Json file include a Blog fact

**[00:40:29]** the directory Eng opiz include a Blog uh

**[00:40:33]** and

**[00:40:34]** fact and photos of shoes from

**[00:40:38]** unsplash all right this is going to be

**[00:40:41]** fun to see what happened so let me give

**[00:40:46]** it let's give it our stores. Json file

**[00:40:49]** and let's watch the magic happen

**[00:41:00]** okay so next what you just see is a

**[00:41:01]** whole lot of things being built and

**[00:41:05]** it'll tell you what's going on over

**[00:41:06]** there this is what's crazy this is

**[00:41:07]** what's changed everything for me you can

**[00:41:08]** just say what you want it to do like in

**[00:41:10]** terms of functionality with users and

**[00:41:12]** logging in you could you could just keep

**[00:41:14]** going and going a director website is

**[00:41:16]** fairly simple for what's possible here

**[00:41:18]** and in future projects I'll be exploring

**[00:41:20]** that myself look it's even making blog

**[00:41:22]** posts here how to make your shoes last

**[00:41:24]** longer the art of shoe repair how do I

**[00:41:26]** know if my shoes can be repaired back

**[00:41:29]** frequently asked questions this is great

**[00:41:31]** I mean when it's this easy it's like why

**[00:41:33]** not why not make a Florida shoe repair

**[00:41:35]** directory you know like why

**[00:41:39]** not all right so it's starting the

**[00:41:42]** application now it tells us what it did

**[00:41:45]** and um I don't know how many tokens it

**[00:41:47]** used apparently not that

**[00:41:49]** many go to the preview here boom all

**[00:41:54]** right let's check this

**[00:41:55]** out find expert shoe repair in Florida

**[00:42:00]** search McFarland shoe repair frequently

**[00:42:04]** asked questions how to make your shoes

**[00:42:07]** last longer okay let's go to McFarland

**[00:42:08]** shoe repair here all right listings oh

**[00:42:11]** it only has one

**[00:42:12]** listing uh is this because of our Json

**[00:42:15]** file did we mess up our Json file no

**[00:42:18]** it's all there um so we need to tell it

**[00:42:21]** to import the rest yeah import the rest

**[00:42:25]** of the schools from this stores. Json

**[00:42:31]** file as you can see here it's just

**[00:42:33]** showing all the listings and you can

**[00:42:35]** obviously go back and forth I'll show

**[00:42:37]** you in a

**[00:42:40]** second this is probably using more

**[00:42:42]** tokens to add in all this

**[00:42:50]** stuff it's populating everything now

**[00:42:52]** there is a faster way to do this which

**[00:42:54]** uh I'm going to show you in a moment but

**[00:42:56]** for now we'll just leave it Let It Go

**[00:42:58]** all right so I put all the stores in

**[00:42:59]** there and now we can preview it again

**[00:43:01]** boom it's got all the listings

**[00:43:03]** here now what we could say is oh we

**[00:43:05]** forgot we could say make the listings

**[00:43:10]** page um show all available cities with

**[00:43:16]** listings and then each City page shows

**[00:43:19]** the entire feed of its listings and I

**[00:43:22]** want to use one column I don't like this

**[00:43:23]** two column thing in one column

**[00:43:27]** now I'm just telling it what to do like

**[00:43:30]** this is how it was for me when I was

**[00:43:31]** working with developers I would just say

**[00:43:33]** hey do this but they were always like

**[00:43:35]** non-native English speakers and

**[00:43:36]** obviously sometimes it's hard to explain

**[00:43:38]** things like this even in the same

**[00:43:40]** language uh so like this is great the AI

**[00:43:43]** just understands what I want and it just

**[00:43:45]** does

**[00:43:46]** it so here we go and as you can see we

**[00:43:49]** still don't seem to have used a lot of

**[00:43:50]** tokens I think we've still Ed less than

**[00:43:52]** 100,000 tokens that's crazy unless it

**[00:43:55]** just hasn't updated it's to me on the

**[00:43:57]** back end n all right so let's

**[00:44:02]** see

**[00:44:05]** starting all right so now we go to

**[00:44:09]** cities and boom M land shows you all the

**[00:44:13]** ones there and so as someone else can

**[00:44:15]** explain better than me this is like this

**[00:44:17]** is a good way of doing

**[00:44:19]** um like SEO Pages it just has all this

**[00:44:22]** all of them on one page and so there you

**[00:44:24]** go we just made

**[00:44:27]** uh Florida shoe repair store now there's

**[00:44:30]** one more step here um obviously you can

**[00:44:32]** keep going back and forth with the AI oh

**[00:44:33]** I finally use 100,000 credits but you're

**[00:44:35]** going to use credits so that's how I

**[00:44:37]** wasted so many credits is I didn't

**[00:44:38]** really realize how simple it would be to

**[00:44:39]** fix things and this is where the coding

**[00:44:41]** comes in or doesn't come in but

**[00:44:43]** essentially for little things or for

**[00:44:45]** bigger things try to get as much done in

**[00:44:47]** Bolt while you're here especially for

**[00:44:49]** like round one you can import a project

**[00:44:51]** back into bolt later but for now I'm

**[00:44:53]** just going to show you what happen so

**[00:44:54]** you have this and you go to export and

**[00:44:56]** you you can download the

**[00:44:59]** file and you can open in stack blit so

**[00:45:02]** you can do both and basically what I can

**[00:45:05]** do now is edit it all so I can say like

**[00:45:08]** okay the title of the page should not be

**[00:45:09]** that it should be shoe repair shops in

**[00:45:13]** Florida save and then I can click create

**[00:45:16]** a repository here connect it to GitHub

**[00:45:19]** and we say shoe repair shops

**[00:45:23]** Florida and I could make a private I

**[00:45:25]** would recommend making a private if you

**[00:45:26]** have data that you worked really hard to

**[00:45:28]** scrape there uh any public repo someone

**[00:45:30]** can just take all the data and do it

**[00:45:32]** themselves all right so let's enable

**[00:45:34]** that bot why not so now we have a total

**[00:45:36]** workspace looks like cursor right looks

**[00:45:39]** like cursor but the difference is you

**[00:45:40]** don't really have an AI agent with you

**[00:45:42]** but you do you can see everything right

**[00:45:43]** next to you as you work if there is an

**[00:45:45]** AI agent in here I don't know how to use

**[00:45:48]** it yeah so there we go now we have this

**[00:45:51]** connected to GitHub I'm going to GitHub

**[00:45:53]** I'm here at Shoop Sho Sho repair shops

**[00:45:55]** Florida and I am going to SSH into it

**[00:45:58]** I'm copying that I'm going to go to

**[00:46:00]** cursor and then I'm going to say new

**[00:46:02]** window and I'm going to say uh clone

**[00:46:06]** repo and I'm going to paste

**[00:46:09]** that boom now it's cloning

**[00:46:13]** everything probably going to ask for my

**[00:46:15]** password yep when you're using SSH

**[00:46:17]** you're going to need a password that's

**[00:46:19]** something you can find all over the

**[00:46:20]** internet but if you have questions about

**[00:46:22]** SSH let me know we're going to open it

**[00:46:25]** here and now we have our site so now

**[00:46:28]** when there's smaller changes or if we

**[00:46:30]** want to use cursor then we just excuse

**[00:46:33]** me cursor AI The Coop pilot and we can

**[00:46:35]** just get in here so this is definitely

**[00:46:37]** stuff for another video but I'll run

**[00:46:39]** through it quickly basically in your

**[00:46:40]** Source components this is how the blog

**[00:46:43]** page will look this is how the city list

**[00:46:45]** City page FAQ Store card will look and

**[00:46:49]** you can essentially adjust everything so

**[00:46:51]** if I want to run this site locally on my

**[00:46:53]** computer so I can make changes then I go

**[00:46:55]** to new terminal and I say mpm install

**[00:46:59]** I'm pretty sure it's mpm yeah what I

**[00:47:02]** would do is I would just um add the

**[00:47:04]** source into here and say how do I run

**[00:47:10]** this project write a read

**[00:47:14]** me boom and then it's going to tell you

**[00:47:18]** what to do run the script similar how it

**[00:47:20]** tell helped us write write and launch

**[00:47:22]** the Python scripts earlier

**[00:47:30]** all right so now it's just going to do

**[00:47:31]** that it's going to tell you what to do

**[00:47:33]** we already did mpm install so we apply

**[00:47:36]** to readme and boom it wrote us a readme

**[00:47:39]** great I already did npm install now

**[00:47:40]** we're going to do mpm

**[00:47:43]** runev and now you can see I'm going make

**[00:47:46]** this a little smaller

**[00:47:48]** here and you can see you can command

**[00:47:50]** click this Local Host link and you're

**[00:47:53]** running Florida shoe rep shoe repair

**[00:47:56]** shops on your

**[00:47:57]** computer okay so now if I make a change

**[00:48:01]** so for example uh header is this um

**[00:48:05]** Florida shoe repa

**[00:48:10]** three yeah Florida shoe directory so I

**[00:48:13]** don't really like that hammer right I go

**[00:48:16]** in here and I could just say all right

**[00:48:18]** here in the header where it says Florida

**[00:48:20]** shoe repair we could say like the number

**[00:48:22]** one Florida shoe repair director yeah

**[00:48:25]** I'll just change it as soon as I save

**[00:48:26]** the file you'll see it uses that hammer

**[00:48:29]** somewhere like right here it says Hammer

**[00:48:31]** so I'm going to highlight it Apple K and

**[00:48:33]** say change this graphic to a shoe oh a

**[00:48:39]** shoe I said show oh yeah okay great and

**[00:48:42]** new I save it and there is no shoe so

**[00:48:48]** what happened I say why um the whole

**[00:48:53]** site doesn't display after changing

**[00:48:57]** Hammer

**[00:48:58]** to now I know why show shoe because it

**[00:49:02]** wasn't imported so up here you see how

**[00:49:03]** it's importing Hammer from lucd react

**[00:49:06]** which is the icon base so we needed to

**[00:49:08]** say import

**[00:49:09]** shoe um

**[00:49:12]** so we accept the file I think I

**[00:49:14]** accidentally can SH

**[00:49:19]** now it also change that shoe oh so there

**[00:49:23]** isn't a shoe icon so um I don't know

**[00:49:26]** scissors

**[00:49:28]** so I just said let's use

**[00:49:30]** scissors I can also just go to the seed

**[00:49:33]** react icons can Google that and I can

**[00:49:35]** see what the icons are like

**[00:49:38]** shoe nothing but whatever we'll change

**[00:49:41]** it to

**[00:49:42]** scissors and there are the scissors so

**[00:49:45]** anything you want to change like text

**[00:49:47]** SEO stuff you can just do that in there

**[00:49:49]** pretty cool and

**[00:49:52]** um yeah there's a lot of possibilities

**[00:49:55]** here and it does to work for you so um

**[00:49:59]** that's it for this video um just to give

**[00:50:02]** you an idea of you know more that could

**[00:50:05]** be possible what I've been working on

**[00:50:06]** here is you go to a listing and you see

**[00:50:10]** the hours you see the map and uh this is

**[00:50:14]** just a map embed that you can tell bolt.

**[00:50:16]** new to put in and you got to give it

**[00:50:17]** your API key but you could say click

**[00:50:20]** here for more or like fix this listing

**[00:50:22]** or submit my own review so these buttons

**[00:50:24]** are going to have added functionality

**[00:50:25]** that help it become more interactive

**[00:50:27]** basically so there's a lot more to

**[00:50:30]** explore when you're building your shoe

**[00:50:32]** repair shop or whatever your directory

**[00:50:34]** is and um I hope this was valuable to

**[00:50:36]** you if you have any questions leave in

**[00:50:37]** the comment thanks for subscribing

**[00:50:39]** giving a like if this was valuable to

**[00:50:40]** you if you have any questions doing my

**[00:50:42]** best to uh help you out sign up for my

**[00:50:44]** newsletter if you want to hear more

**[00:50:45]** about what I'm working on and the joy I

**[00:50:48]** find in building and putting things

**[00:50:50]** together thanks for joining me today

# Full Text (without timestamps)

today in this video I'm going to show you how I built montor find.com in just 7 Days basically it is a directory website that will be SEO optimized and it has 7,000 listings thereabouts and I scraped a whole bunch of data from the internet and I did this all basically with AI tools turned around a pretty nice little website so I think anyone can do this these days if you're remotely competent online I mean if you even know what SEO means I'm sure you're competent enough to figure this out we're going to use cursor Ai and bolt. new so without further Ado let's get started first I'll just show you the website and what we're looking at basically you can go to any state in the Union and you can find a Monas School a list of Monas schools as you can see it has the the name of the city the name of the State uh and then we've got reviews that are scraped from Google Maps so here's a four a five star review um out of four total I might have to clean that up it's got a map embed here with the proper instructions it's got a link to Google Maps and it's got phone number it's got it so all this these descriptions in every single uh listing were scraped with python script open source called crawl for AI which we're going to get into and then an llm an open source llm was used to make these descriptions um pretty fun project overall here's the project window in cursor as you can see we've got a big Json file full of all the listings here which will converted from a um from a CSV we've got a Blog in here as well um and a pretty fairly easy system for making new SEO blog posts and then there's a frequently asked questions which in and of itself I think is a bit of a masterpiece though I can't take full credit for it because Claude AI really helped me the first thing you need to do which I won't be covering in this video is find your lucrative Niche find the right search term which doesn't have a lot of competition check out Frey ch's video shout out to Frey who inspired me to build this so we're going to focus on just the build process so once you have your topic then you're obviously going to want to scrape listings on Google Maps we're focused on local listings here that are more Evergreen that people would need in their like neighborhood so first we're going to go to Google Maps and we're in Miami area here it's already searching for monor schools let's just come up with something new let's say we're looking for um shoe repair shops in Florida let's just say Florida McFarland shoe repair okay shoe repair okay so we got some results here we're going to make a shoe repair website shoe repair directory all right for in his YouTube video recommends using a website like outscar where you can actually it's kind of cool cuz you can pay with Bitcoin um you pay PayPal whatever you want but it's got a Google Maps scraper basically so you so we would say Sho uh repair shop great shoe repair shop and we want to say just Florida here so we're choosing this Florida all right unlimited results enhanced results with other services we don't need any of that in some cases it might come in handy but contains one of sh repair shop delete duplicates use zip codes all right so we're going to click get data it's going to tell you how much money it's going to cost from $9 to $61 um I'm not going to pay for that but you can if you want instead I'm going to show you how to use a script on your own computer that just uses Google API credits that you actually get free I think $300 free credit when you sign up at the time of me recording this so you can technically actually get it all for free yeah so you can see here I've got $141 of free credit so we're going to use that to scrape this data okay so first things first is you're going to want cursor cursor is an AI coding app uh it's a lot like Visual Studio code where basically you're going to have all these files over here you're going to have access to your git repository lots of plugins but there's an AI agent over here that helps you do anything the agent will build the ask agent normally uses a different model and uh it just helps you figure things out um I have never really used edit yet but it's probably for editing software but the agent helps you build so we're here we have nothing so I'm going to make a I'm not even going to do anything yet I'm going to say write a script that uses Google Maps API um I think it's called places um to scrape Google maps of all listings in the State of Florida for the term shoe repair shop export the results as a CSV file which includes one rating and then I'm Al okay so let's just do that okay one how about we say this one positive rating and one negative rating so maybe you want more ratings um the ratings is something I'm going to integrate a system into monor find later but for now it's good to start with ratings and we're just doing a sample video here so I'm just going to do one rating so Watch What Happens I send that okay so it shows python when I did mon story find I used JavaScript but that's okay python ends up doing a lot of the data scraping later when we want to generate the descriptions so as you can see over here it's literally coding everything I just said I want it uh and it gets even crazier later when we're going to use bolt to then plug in the data we scrape into a website and it just makes a website so this is all still the back end we can't use bolt yet so here it says make a new file requirements. text and you can either press this check mark here accept or just uh Apple y command y or just accept boom uh scrape sheet repair okay now I made a python script here it is this in green shows you what it added and then later when we edit you'll see things in red things in green red means it's cutting it green means it's adding it so it just wrote this whole script um now you can obviously vet the script and look through it and if you really want to learn how to understand coding then you simply go in here and Google everything you don't understand or you copy it and you say what does that do but let's do that in a minute so I can I can just copy it and say like what if I don't want what if the AI is just putting malicious code on my computer um I'm going to add it to the chat and I can say what does this do and as you can see here it knows lines 36 to 44 which is what I highlighted that's its context and then me explains what it does now I Ed the agent for that I should have used ask if anything it'll just save the credits but whatever all right so then it tells me exactly what this does it starts from Florida Center coordinates latitude blah blah blah and so I just copied these lines right here there's obviously a whole script here so if you just go through the whole thing you can actually learn what each thing does you don't need to be a coder anymore you just have to learn how to tell it what to do so anyway I'm going back to up here and it's giving me the script and I say accept file boom it accepts it and it'll save it automatically and then you'll notice here it's loading environmental variables we're going to need one of those files as well because we're going to need an API key uh and then a read me so we say yes to the readme okay so then it gave us everything and it's cool is in that readme it tells us what to do it says Okay first install the requirements then you create a n file with your Google Maps key so I'm going to copy that I'm going to make a new file right here call it. EnV boom now I'm not going to show you this screen again but basically I'm going to paste the Google Maps key right there so we simply go to Google Cloud I've got my monor Schools project here you're going to have to make a new project you can Google it it's fairly straightforward go to apis and services go to credentials and you're going to create credentials you're going to create an API key and then it's going to give it to you and you're going to paste it into your end file now if you've never coded before you might have to install python on your computer or JavaScript or whatever cursor asks you to do in this case I already have Pip installed uh if we look at a command line pip means like what software we're using pip or python or node npm that's like the general like what language does it need to use to launch the script and then this is the command install and then boom it's it's going to install whatever it reads from the requirements file so we're going to open a terminal new terminal we want to run pip install requirements now we're going to have an issue here maybe because on my computer like we'll definitely have an issue here with python script Sho repair we're going to need to use Python 3 Let's see looks like it worked okay so it's installing all that if you get an error you copy it and then you either send it to the chat or you Google it and then it'll tell you what to do this is a normal part of coding and processing something will always go wrong and we need to be confident and Sovereign enough in our abilities to figure out the issue it's easier than ever now thanks to llms so I'm making this I'm making this video now cuz I'm confident doing it and I think you can be confident too just takes experience we only get experience by building okay now we have an error this is a lot so see how it even gives me this button add to chat see how it's got the context here the lines and I just say what happened not fully compatible let me modify the requirements so basically it knows I'm using python 313 but it's not compatible with that version of pandas which it needs so it's changing the requirement the dependency here I accept it I save it and then I simply press up uh on the terminal and it's going to go back to that pip install requirements and I press enter to run it let's give that a minute it looks like it happened fast boom there we go we got it now the next thing we got to do if we scroll back up we can run the script so if I try to run it now it's probably going to give me an error because it doesn't have python but it does have Python 3 so it's scrape shoe repair. python okay now let's do it now see how it's got some it's called Uh I don't even know it's called verbose logs in there so basically it's kind of telling you what it's doing data exported okay it found 24 shops great that probably wasn't very expensive hopefully it wasn't very expensive so it gave us something we haven't looked at it yet and we don't know if it did it right so this is what's really important as as we get AIS to do things for us especially if we don't know how to do them ourselves we don't know if they're going to work right but that's why we do step- by-step in iterations so let's take a look at the CSV file looks like it made all these columns in fact let's just open this in a CSV Viewer okay so I hope this isn't too small we've got name address rating total ratings latitude longitude positive review negative review if they got one the CSV looks good but I'm realizing we forgot a couple important things mainly the website so I'm going to exit that out and I'm going to say add a column for website and description um into the into the scrape shoe repair script okay so we're it gave us his changes let's add those let's say yes cool accept and then say we also need to add a zip code separate full address street address ZIP code and city into their own unique columns okay so even if we realize we need more data here this is just what it's going to be because this is just a tutorial video but as you can see we can keep enhancing this script in the beginning we obviously don't want to use too much data to for our apri credits to pay money until we know exactly what we want I accept that it's got street address city state ZIP code full address okay great um okay so it's doing everything we want now can we also scrape for a description of the listing now we are going to use an llm to scrape the listings website but this is also a good move just in case like Florida Sho repair how many websites do they have I mean maybe they have lots of websites or maybe they're these old mom and pop businesses that just run off of board and mouth and they don't think they need a website and they don't cuz their business is still going um we're going to try both ways here so we're going to accept the accept there okay okay so now we're going to run this script again all I got to do is press up python scrape shoe repair I'm going to open up our original CSV and just delete it just so there's no potential issues there Python 3 scrapes shoe repair. piy boom okay it worked so now after few back and forths we're getting the address the full address the country the state the city the ZIP code we got the phone number we got the URL and we got a negative review and a positive review great so now we have to work with and we got 42 listings that's like perfect actually it's less than that I think it's 24 24 listings great little mini tutorial website so the next thing I'm going to do uh I'm going to clear my files out here a bit I'm going to make a new file and I'm going to call it organized data. piy and I'm going say in this file and I'm going to tag it organized data. py write a script that removes listings from our CSV file that has no address or no state if there is no website um add no website uh but where are we going to add it so let's do this um add no website yeah to the website column let's just add that there for now so what that's going to do is just parse everything and say hey this one doesn't have an address this one might not be real so just remove it and I'm going to rename our CSV file to just data. CSV so it's easier all right so now it's writing an organization script to just filter out if anything's broken uh right here it says hey you know choose your CSV file what I like to do so we can use this again later is send it to the chat and say make it so I can choose the CSV file in the command line and the output file is simply file name updated. CSV now we got organized data and it says to run it all we do is say organized data your your input file to CSV so I say Python 3 for you it might be python for me it's Python 3 we're going to run organize data. piy and we're going to data. CSV boom so now we got data updated over here and um let's do this again let's add a success message that shares how many um rows were updated or deleted so we don't even know what happened so let's add in that message and what you can do also the next level if you have a lot of listings it say send all the failed ones to a new CSV file so I can just check them manually and that is a step that I'm still working on and organizing my data from monor find okay so it added this new stuff let's run it again now it says updated entries four so to four entries it added no website perfect so we're going to go to Swift Shoe Repair Inc because it said no website okay there's Swifts um and it's true doesn't look like it has a website um bag clinic and shoe repair does look real and open this is how we enhance the data you know we just check it out 5 years ago when were these reviews 3 months ago okay so it's still there um but then I would just go through each one of these ones with no URL and um and just check to see if they're still around boom um then we see missing address is none address without Florida none okay so there we have our file it's clean depending on your data you might have a lot to do here and it might take a while um or you do it in a future round but in general I try to at least get out like the dead ones before we even go live so now we have our data here okay data updated. CSV and now we're basically ready to start a brand new project here okay so we've got our base data but now we need to make a description in order to do that we need to scrape the URLs of each website in here go through it find the important information and then generate a description from it so for that we're going to use an llm AI and we're going to use software called crawl for AI which is an open source llm friendly web crawler and scraper and and it looks a little intimidating it was for me in the beginning but it actually is going to make a lot of sense as we go through it so first things first we are going to make a fresh file and we'll just call it description generator. Pi okay now the first thing we need to do if we just make a quick list we need to connect to the LM AI which is going to be Venice and then we got to uh so see and cursor it gets you all this stuff from the GetGo um but it doesn't know that we're not really programming here then we need to uh scrape the website for two to three paragraphs of valid information valid text and then we're going to generate a description of the website and we're going to Output the description to the column uh to the description column of the CSV file we're going to repeat the process save the updated yeah there we go um okay so that's what we're going to do here and we're actually going to copy and paste that into the agent in a minute but let's just first say okay how are we going to scrape the website well we're going to use crawl for AI and we're going to use um what's basically called llm extraction method which is llm strategies here why would you use an llm for scraping well it's going to help structure the data if it's unstructured the website itself it can reason through it and be like okay that's necessary that's not necessary just in general why do we use l l m because they help us do tedious things faster and this is very definitely a tedious thing so the first thing I'm going to do I've opened the agent here and we're going to start telling it what we want to do here but the reason I made these steps is that it can be a little complicated here so the reason I made these steps is because this can be complicated to tell the agent to do it all at once so we're basically going to test it step by step until we know each step is working so then by the time we run it we know it can do what we want we could just say hey do this and it just sends it all of there we could just say hey do this and paste that all in there but the truth is probably not going to work very well so we're going to start from the top first thing we're going to do here is go is connect to the llm AI so I'm going to actually make a new document just going to call it steps. MD or even steps. text we have the steps we want to do so we are going to write a script that's tell it what it's called it's going to be called um description generator. piy that scrapes URLs in our data updated. CSV file for information to summarize into a brief summary first let's connect to the Venice AI API now we can't just say that with open AI maybe anthropic yeah but Venice isn't as deep into the system so we're going to go to Venice ai's documentation and in here we're going to go to their reference and basically I don't know what a Swagger definition is but in fact this is actually perfect we're going to take that Swagger that yaml Venice actually let's make a folder called Venice docs okay and in Venice docs we're going to make a file called Swagger DOL going to paste all that in there and that has like all these little things that the script is going to need to know in order to communicate with the Venice aai through their API including the URL that might be enough um but basically I'm going to copy all this in there too and I'm going to say what do we call Basics Mt copy that in there um all right so now we've copied and pasted whenever you're going to use an API or a script which we're going to use with craw for AI copy and paste it into your workspace in a file so that way you can drag this in and say okay write a script we're going to write a script but first let's connect to the Venice AI API successfully check Venice docks and add it to the python script all right so now I hadn't saved that but hopefully it's okay so I've already got the Venice API key in there and that's giv me the code and I'm going to apply that to description generator except file now you can see it opens open AI cuz it uses open ai's uh module and you just change a few things to Venice looks like it's got the right base URL um I've been working with Venice AI enough to know that this is working uh so okay we're going to save that and now we are going to install these dependencies okay they're all installed now now we're going to do Python 3 description generator. py um it failed to do everything but it looks like we didn't it yeah so add a command so we can just test the API connection all right so now it's just going to add this little this um this text basically so once we connect it says hey it's work and we just have to run Python 3 description generator test so this flag here is just saying hey we're going to test it that you can add like okay it it failed so we're going to send this what happened and our lovely AI agent is going to tell us what happened and let's see okay so it's going to change whatever it needs to change save it let's try that again testing connection successful all right so that's step one all right now let's go back to our steps and we got okay so that one's done two scrape the website all right so here's where crawl for AI comes in so crawl for AI here is a really awesome script here and basically as we're going through we can obviously ask the AI to write a code for us but code creation is not simple there's a reason it's a high paying career F you know it's it's a big deal it's been a big deal for so long cuz it just takes time things don't always work right even if you're in AI so while it is getting better it still is great to use project repositories Frameworks out there that already built for you and then the AI doesn't have to build it it just says oh use that use that use that so what we're going to do is we're going to install we're going to do the quick start here and we're going to install and then we're going to implement crawl for AI into our project that's installed setup boom and then now it says verify your installation make sure sure it's working that's another python script going to check through example.com it's going to pull 300 characters so I'm just going to copy that I'm going to make a new document called test. python put that in there save it and then just say Python 3 test. python it's going to test this connection to example.com and it worked got example domain this domain is that that's what it that's what it scraped all right so craw fre ey is open so back to our steps so we want to scrape the website and generate a description so first off scrape the website so next we want to scrape the U the website URL from the from the listing Row in data updated. CSV and we're just going to say find two to three paragraphs of relevant information to summarize the listing we can even say summarize the shoe repair store format it to markdown let's just say prune filter it and then format it to markdown so that's the first thing we're going to do um basically and I I know that process because I've been using it and I'm going to walk you through it here now I recommend you go through all of this especially if you're going to do a lot of data scraping but basically you copy and paste what you want from the read me just like we did with the Venice API specs copied it into a document you do the same thing here for this script so then you you can just say hey use browser config module use crawler run config module or if you're doing deep crawling use uh use uh deep crawl strategy llm web scraping strategy so instead of just blindly telling the AI to do it you give it the documents so you go to GitHub you just download the zip and then I copied the docs folder into cursor here and then here in mdv2 that's really where we see everything so I'm going to document I'm going to drag mdv2 which are the docs and I'm going to put that uh what needs to be higher up Okay C documentation in mdv2 now one thing we need to know though is what modules do we want to use use modules all right so first off we're going to that's simple crawling it's basically going to do that on its own we're definitely going to use async web crawler because that's the very basics of everything um async web crawler we going to need crawler result I believe um yeah cuz this results everything it puts out everything it finds um llm config um we're going to use LM config we're going to use markdown generation to clean it up uh we're going to use fit markdown and in order which is basically using a Content filter like we mentioned so we're going to use pruning content filter here once again I'm very familiar with this but you might have different purposes like bm25 could be good if you want to scrape search results for example so make sure you go through all this for your own use case I'm just doing kind of the the be Basics here we do want to make a cach so cach mode that'll kind of speed things up a little bit all right so I think that's good we'll just say modules to consider all right so it's not going to generate it yet it's not going to generate anything yet it's just going to scrape the website and get the two to three paragraphs of text how about we say format it to markdown create a unique directory per listing with its own markdown file so it has the raw data we got the two to three paragraphs okay so we're going to send that out now and we're going to see how it does and once again it has the documentation here for the script doesn't mean it's going to do it right but it looks so so far like it's doing it right and if we said connect to vsai and now scrape the website and now generated description all at once it wouldn't work in fact I recorded this for an hour before and I did that and it didn't work so I'm recording it again um so that's a very important lesson I'm trying to pass on to you there are the changes here you're seeing it's importing more modules it's added all this code okay accept file okay so let's see what happens here we got pyone 3 the last thing we want to see is uh where is it getting yeah it's getting the right CSV file Python 3 description generator. py okay so we got an error there um we highlight that we add it to the chat what happened uh we switched to ask for that basically ask I believe you use like a cheaper llm and then agent you use a more powerful more expensive llm smarter one all right so description generator all right so whatever I don't know what it did um oh you have to do it somewhere else whatever that is okay let's see still not working let's send it to the chat what happened oh default markdown generat in a different module so yeah I mean the AI has to figure stuff out just like us humans so the simpler you make and the more concise all the directions the more chance it'll be done accurately the first time but of course you can okay so the the script ran but of course you can always just do it like this and just debug as you go um all right so we got an error there let's send it here happened prob URL right it didn't have anything to crawl still didn't work we're going to say error again are we sure it's pulling the URL from the data updated. CSV file he so this part of the video is kind of boring cuz I go back and forth a bit so I'm going to cut back in So once we solve the problem of it finding the URL we go on to the next problem the key Point here is solve one problem at a time don't try to solve multiple problems with one prompt it will just make more problems probably okay wait all right it's crawling it's crawling okay we still failed but we got some progress crawl result as no okay so uh oh let's see we see that at least it is scraping data it's getting the original stuff yeah so there we go that's probably like part of the description quality so it's doing the scraping right that much is working but what's not working is whatever is next so yeah generating a description so not working it's something to do with the Venice API we need to remove Venice parameter since we're using the async open AI client fix the generate okay yeah so basically this is just slight things to do I think between open AI versus Venice AI apis let's see let's see oh I think it worked processing Row one all right so let's cancel that and let's just see we should have updated file here it generated a description we just didn't tell it where to put it and that's perfect because all we wanted to do is scrape website now it is generating the description so we've got the description so now we need to say um make make sure the script make sure the script outputs the description the generated description to the description column of the uh of an updated CSV file call that data updated uh descriptions that's CSP and we're almost done I think it's already doing this these last two might already be built in we're going to find out okay so there's our code we'll accept everything there and I've got a good feeling that we're we're getting close boom so it saved the markdown file not seeing a new CSV file I have a feeling it's doing it in the end which is why we wrote these steps so it's probably just putting the descriptions it's generating into its memory which might be adding extra context yeah and then it makes the file at the end exactly what it did here so there's a brief description all right so let's open this it's back here where is it right here descriptions all right now we obviously need to work on this prompt here's a brief description of the shoe store um yeah it's not working but it's close Okay so at least it's like doing what we want it's just not doing it right it skipped the no website on that's good they did have this here so that's the description I gave first Sho repair also known as America's gobler offers expert shoe repair okay I mean they do it too here so all right so it's good but it's no cigar so we're going to say um we don't even need to say this to the agent let's just go to the prompt and let's find you are a helpful assistant that creates concise informative descriptions of sh repair businesses so this is the prompt it's giving the llm focus on key Services Specialties and unique features um okay and then we just say um do not output anything else with the description thank you autofill um and we all say do not make anything up if there's not enough data say not enough data okay so now we change that prompt and um let's just say create a two to three sentence description all right so now I'm going to delete that new CSV file so we don't get all messed up and we're going to run this new script again if this is happening how it should be which I'm not sure it is but it's got these content MD files already there that's just generating the descriptions from there's one other thing we need to do still which is actually steps five and six together which is well it's already repeating the process but then save the updated CSV file as it goes so let's save the rows with descriptions to the new CSV file one by one instead of waiting so we're just going to have to do that and then we're done with the steps document um once again we go to the descriptions here and let's just see how that line did there we go McFarland shoe repair also known as am's cobbler offers high quality shoe repair services Drive shoe repair not enough data Phenix muled shoes inserts and braces Redwing shoe store in Lakeland Florida okay so we got it okay the last thing we would do like I said we had say hey write it one by one instead of all the time at the end so if uh we delete that right now and do that again we're going to notice it makes a new CSV file sooner boom so it's there then it's got uh yeah same description pretty much and it's just going to add our new ones there one at a time perfect so we have some base data now but we're going to want to clean it a little more which is why like on this one not enough data let's go to this website let's see I mean yeah I see why there's not enough data huh um but it's it's there right it's not actually a shoe store they fix shoes they kind of fix shoes but we're looking for people I'm just going to delete that that listing uh not enough data let's check this one there's a Facebook link that would be why now maybe in another video we'll go over how we can give it our Facebook login and like hey log in and get this information I'm not going to do that in this video um all right so we're running it one last time we're going to have our final data we're going to delete a few listings and next up we're going to put it into a website all right so it's making the new folder made the new CSV file boom we got it and it's got descriptions okay great so we have our data updated descriptions. CSV okay so we got this there's one last thing we want to do and that is take the address and split it into a few different columns CU we want to have the city in its own column the state not so much cuz it's all only just Florida and the zip code so let's say let's write let's make a new file and let's call it um blit address and we're going to say write a script as split address that separates the street address from the city Z code State and Country into new columns all right so here we go split address. piy um let's see what CSV file is it using data updated descriptions that's CSV Alpha file let say data updated addresses okay fine so this is where your taxonomy it's getting important you're going to get all these data files you got to stay organized I just keep making new folders after a new uh leap okay so we're going to run that now well run command all right cool so it worked let's see now let's see we got address street address city state ZIP code country go all the way to the end D address city state ZIP code country perfect so now we have all the data we need now what we're going to do is convert it to Json convert to json. Pi and we're going to say we could just put it here say write a script that converts data updated addresses that's our latest one to Json for use in populating a directory website to a Json file so CSV is great for spreadsheets uh it's good for a lot of things it could also work on a website but Json is the standard so here we go we're going to run this script now we're going to convert our lovely uh going to convert our lovely CSV file Python 3 convert json. piy boom and now you'll see what it looks like is is stores. Json oh shoe stores great so now we got stores. Json and it's organized all that information this is what a Json file looks like each new squiggly line means it's a new entry boom latitude longitude great we got all the info we need now make our shoe repair shop in Florida hey guys before we continue with the best part of this project making the website I just wanted to let you know that this is a brand new channel and if you're still watching if you're finding value in this I would absolutely love it if you could press like and maybe even subscribe if you want to continue building with me I'm going to be making videos like this a lot I love building I love using AI so if you could do that that would be really awesome okay let's build this website we're going to go to bolt. new there's also bolt. DIY which is a fork of bolt. new uh and you can run it locally with local models but it is pretty slow so um I I'm using bt. new um I love open source stuff but I tried both on DIY it just took forever so I have two two two million monthly tokens remaining I'm going to try to use less than half of that I used 8 million for building monori fine uh and I was learning bolt. new as I went so I realized I could have done it for cheaper I'm going to say build a direct I'm not even gonna okay you could use versel nextjs you could study up on these Frameworks spelt um and choose which one you want but we're just going to let it do it okay um it's going to use nextjs probably um but I'm going to show you two ways to do this first we're just going to give it to bolt. New build a directory uh for shoe repair shops in Florida uh the listings are in the attached Json file include a Blog fact the directory Eng opiz include a Blog uh and fact and photos of shoes from unsplash all right this is going to be fun to see what happened so let me give it let's give it our stores. Json file and let's watch the magic happen okay so next what you just see is a whole lot of things being built and it'll tell you what's going on over there this is what's crazy this is what's changed everything for me you can just say what you want it to do like in terms of functionality with users and logging in you could you could just keep going and going a director website is fairly simple for what's possible here and in future projects I'll be exploring that myself look it's even making blog posts here how to make your shoes last longer the art of shoe repair how do I know if my shoes can be repaired back frequently asked questions this is great I mean when it's this easy it's like why not why not make a Florida shoe repair directory you know like why not all right so it's starting the application now it tells us what it did and um I don't know how many tokens it used apparently not that many go to the preview here boom all right let's check this out find expert shoe repair in Florida search McFarland shoe repair frequently asked questions how to make your shoes last longer okay let's go to McFarland shoe repair here all right listings oh it only has one listing uh is this because of our Json file did we mess up our Json file no it's all there um so we need to tell it to import the rest yeah import the rest of the schools from this stores. Json file as you can see here it's just showing all the listings and you can obviously go back and forth I'll show you in a second this is probably using more tokens to add in all this stuff it's populating everything now there is a faster way to do this which uh I'm going to show you in a moment but for now we'll just leave it Let It Go all right so I put all the stores in there and now we can preview it again boom it's got all the listings here now what we could say is oh we forgot we could say make the listings page um show all available cities with listings and then each City page shows the entire feed of its listings and I want to use one column I don't like this two column thing in one column now I'm just telling it what to do like this is how it was for me when I was working with developers I would just say hey do this but they were always like non-native English speakers and obviously sometimes it's hard to explain things like this even in the same language uh so like this is great the AI just understands what I want and it just does it so here we go and as you can see we still don't seem to have used a lot of tokens I think we've still Ed less than 100,000 tokens that's crazy unless it just hasn't updated it's to me on the back end n all right so let's see starting all right so now we go to cities and boom M land shows you all the ones there and so as someone else can explain better than me this is like this is a good way of doing um like SEO Pages it just has all this all of them on one page and so there you go we just made uh Florida shoe repair store now there's one more step here um obviously you can keep going back and forth with the AI oh I finally use 100,000 credits but you're going to use credits so that's how I wasted so many credits is I didn't really realize how simple it would be to fix things and this is where the coding comes in or doesn't come in but essentially for little things or for bigger things try to get as much done in Bolt while you're here especially for like round one you can import a project back into bolt later but for now I'm just going to show you what happen so you have this and you go to export and you you can download the file and you can open in stack blit so you can do both and basically what I can do now is edit it all so I can say like okay the title of the page should not be that it should be shoe repair shops in Florida save and then I can click create a repository here connect it to GitHub and we say shoe repair shops Florida and I could make a private I would recommend making a private if you have data that you worked really hard to scrape there uh any public repo someone can just take all the data and do it themselves all right so let's enable that bot why not so now we have a total workspace looks like cursor right looks like cursor but the difference is you don't really have an AI agent with you but you do you can see everything right next to you as you work if there is an AI agent in here I don't know how to use it yeah so there we go now we have this connected to GitHub I'm going to GitHub I'm here at Shoop Sho Sho repair shops Florida and I am going to SSH into it I'm copying that I'm going to go to cursor and then I'm going to say new window and I'm going to say uh clone repo and I'm going to paste that boom now it's cloning everything probably going to ask for my password yep when you're using SSH you're going to need a password that's something you can find all over the internet but if you have questions about SSH let me know we're going to open it here and now we have our site so now when there's smaller changes or if we want to use cursor then we just excuse me cursor AI The Coop pilot and we can just get in here so this is definitely stuff for another video but I'll run through it quickly basically in your Source components this is how the blog page will look this is how the city list City page FAQ Store card will look and you can essentially adjust everything so if I want to run this site locally on my computer so I can make changes then I go to new terminal and I say mpm install I'm pretty sure it's mpm yeah what I would do is I would just um add the source into here and say how do I run this project write a read me boom and then it's going to tell you what to do run the script similar how it tell helped us write write and launch the Python scripts earlier all right so now it's just going to do that it's going to tell you what to do we already did mpm install so we apply to readme and boom it wrote us a readme great I already did npm install now we're going to do mpm runev and now you can see I'm going make this a little smaller here and you can see you can command click this Local Host link and you're running Florida shoe rep shoe repair shops on your computer okay so now if I make a change so for example uh header is this um Florida shoe repa three yeah Florida shoe directory so I don't really like that hammer right I go in here and I could just say all right here in the header where it says Florida shoe repair we could say like the number one Florida shoe repair director yeah I'll just change it as soon as I save the file you'll see it uses that hammer somewhere like right here it says Hammer so I'm going to highlight it Apple K and say change this graphic to a shoe oh a shoe I said show oh yeah okay great and new I save it and there is no shoe so what happened I say why um the whole site doesn't display after changing Hammer to now I know why show shoe because it wasn't imported so up here you see how it's importing Hammer from lucd react which is the icon base so we needed to say import shoe um so we accept the file I think I accidentally can SH now it also change that shoe oh so there isn't a shoe icon so um I don't know scissors so I just said let's use scissors I can also just go to the seed react icons can Google that and I can see what the icons are like shoe nothing but whatever we'll change it to scissors and there are the scissors so anything you want to change like text SEO stuff you can just do that in there pretty cool and um yeah there's a lot of possibilities here and it does to work for you so um that's it for this video um just to give you an idea of you know more that could be possible what I've been working on here is you go to a listing and you see the hours you see the map and uh this is just a map embed that you can tell bolt. new to put in and you got to give it your API key but you could say click here for more or like fix this listing or submit my own review so these buttons are going to have added functionality that help it become more interactive basically so there's a lot more to explore when you're building your shoe repair shop or whatever your directory is and um I hope this was valuable to you if you have any questions leave in the comment thanks for subscribing giving a like if this was valuable to you if you have any questions doing my best to uh help you out sign up for my newsletter if you want to hear more about what I'm working on and the joy I find in building and putting things together thanks for joining me today