Func-AI Tutorials and samples

1

Introduction to Func-AI - Unleash Your Data Science and AI potential

Welcome to the world of Data Science and AI! In this comprehensive set of training tutorials, we'll embark on a journey to unlock the full potential of this powerful software, enabling you to work smarter, not harder. Whether you're a seasoned professional or a newcomer to the data science and AI realm, this tutorial is designed to cater to your needs.

Why Func-AI?

Func-AI isn't just another AI chat tool; it's your expert partner and co-pilot in data and AI competency. It's a feature-rich, user-friendly application that can fast track your data analytical and ML capabilities, and bring your deeper, faster insights to your projects. Whether you're an analyst looking to efficiently code in Data Science languages, an expert looking to scale and share your knowledge, or a complete beginner looking to learn and understand more, Func-AI is here to make your journey smoother and more rewarding.

What Will You Learn?

In the upcoming modules, we will explore the core features and functionalities of Func-AI, providing you with the knowledge and skills to harness its full potential. Our tutorials will cover:

1. **Getting Started:** We'll guide you through the installation and setup, ensuring that you're ready to dive into the software with confidence.

2. **Basic Functions:** Learn the essential tools and functions, such as creating, saving, and exporting files, to get you up and running quickly.

3. **Advanced Features:** We'll take a deep dive into the more sophisticated capabilities of Func-AI, from demystifying complex tasks to leveraging shortcuts for increased efficiency.

4. **Best Practices:** Discover tips and tricks from experts in the field, allowing you to make the most of Func-Ai and Python Notebooks in your everyday work.

5. **Real-World Projects:** Put your knowledge to the test with hands-on exercises that simulate real-world scenarios, solidifying your understanding and enhancing your skills.

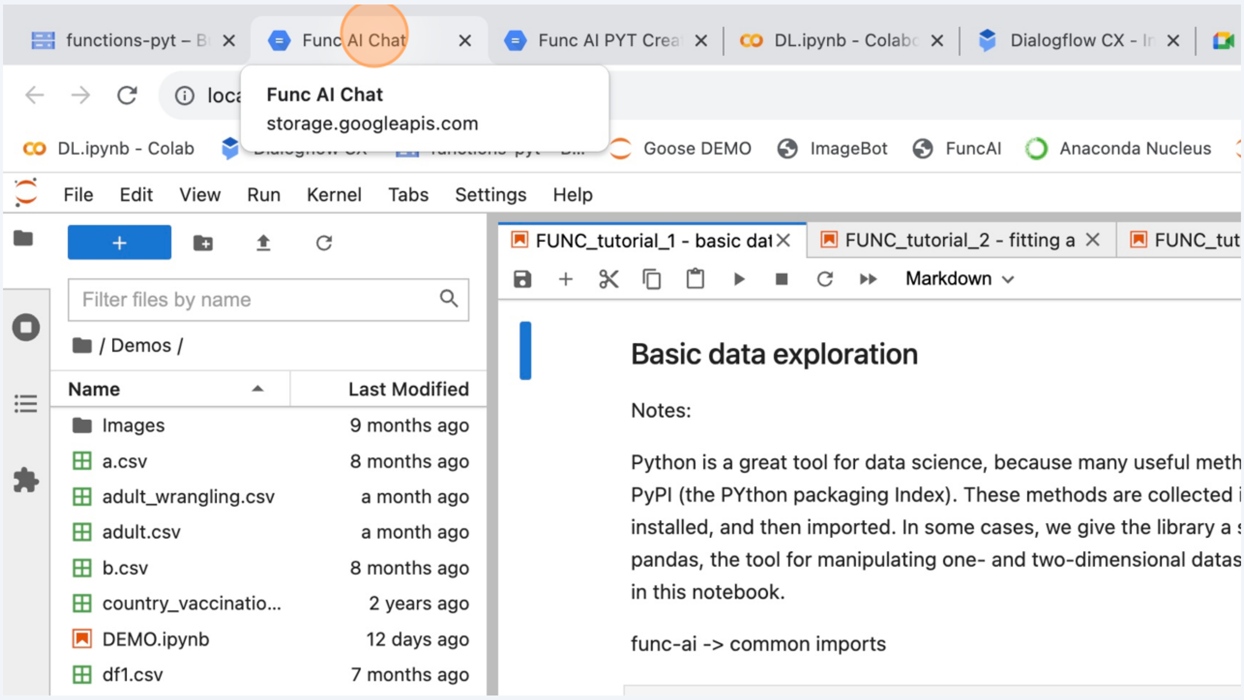
By the end of this tutorial, you'll be well-equipped to harness the full potential of Func-AI, boosting your productivity and making your projects shine.

So, without further ado, let's embark on this exciting journey together. Func-AI awaits, and it's ready to help you accomplish more than you ever thought possible. Let's get started!

Tip! Explanation of the training guide. To keep the flow as straightforward as possible, we will start with an explanation and guided tutorial for the first exercise, then fast track the remaining steps. After all, Func-AI is designed to be quick and intuitive to learn!

2

Open your Chrome Browser and navigate to your Jupyter Notebook using the URL supplied by your IT support team.



Name	Last Modified
Images	9 months ago
a.csv	8 months ago
adult_wrangling.csv	a month ago
adult.csv	a month ago
b.csv	8 months ago
country_vaccinatio...	2 years ago
DEMO.ipynb	12 days ago
df1.csv	7 months ago

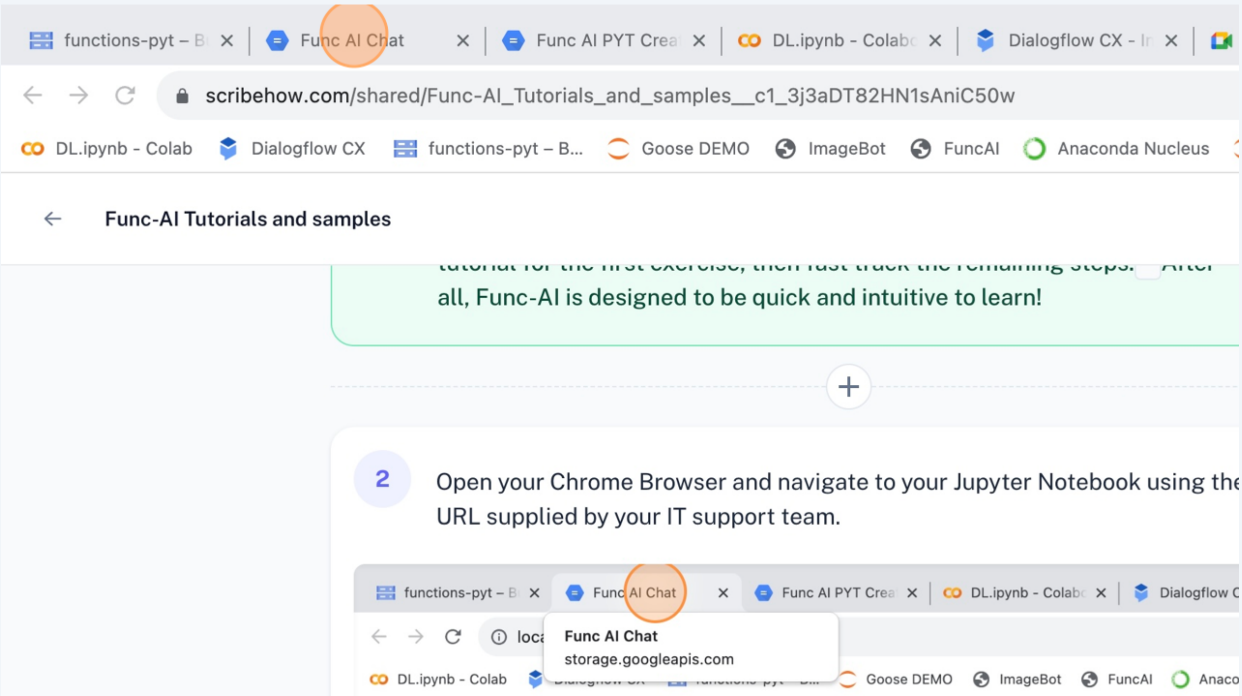
Basic data exploration

Notes:

Python is a great tool for data science, because many useful meth
PyPI (the PYthon packaging Index). These methods are collected i
installed, and then imported. In some cases, we give the library a s
pandas, the tool for manipulating one- and two-dimensional datas
in this notebook.

func-ai -> common imports

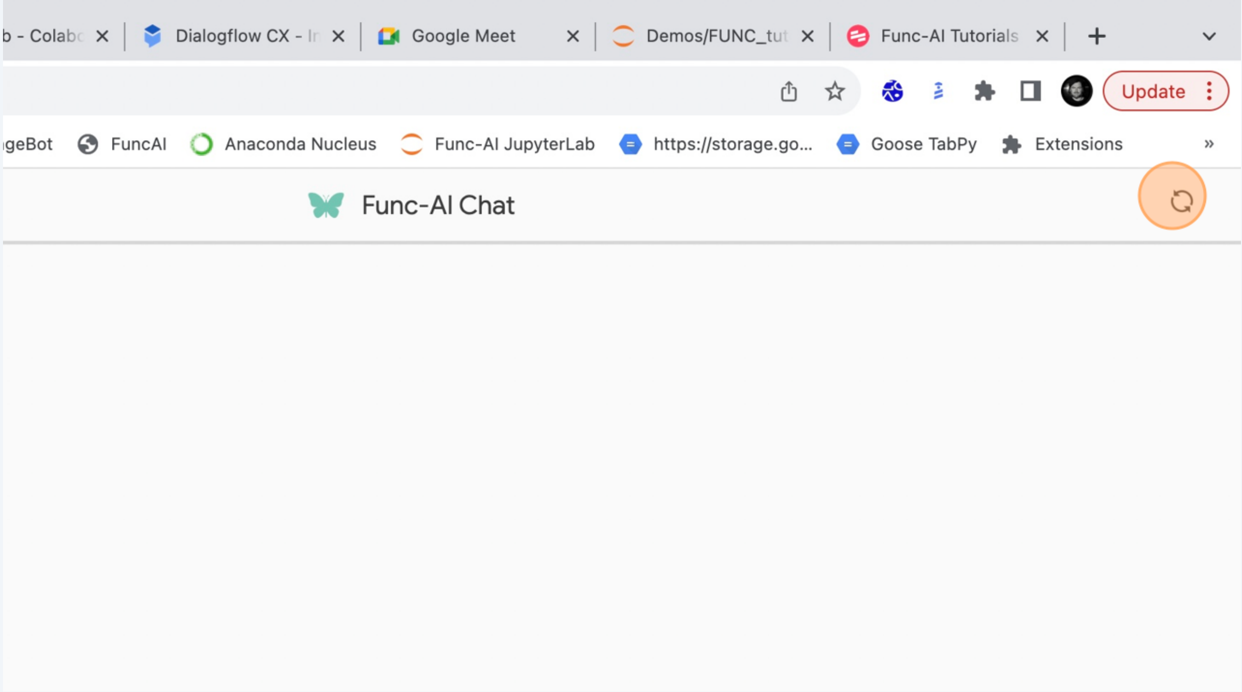
3 Click the Func AI Chat tab



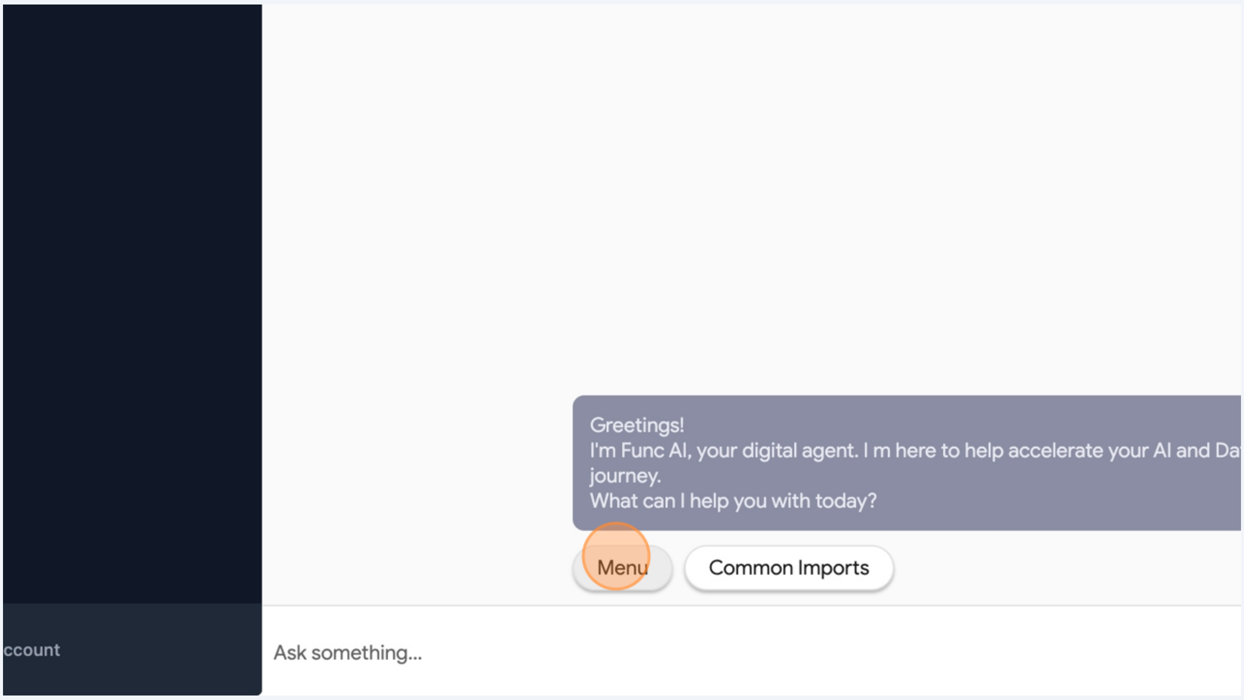
all, Func-AI is designed to be quick and intuitive to learn!

2 Open your Chrome Browser and navigate to your Jupyter Notebook using the URL supplied by your IT support team.

4 Your Func-AI Chat Browser now opens and you can begin your conversation!

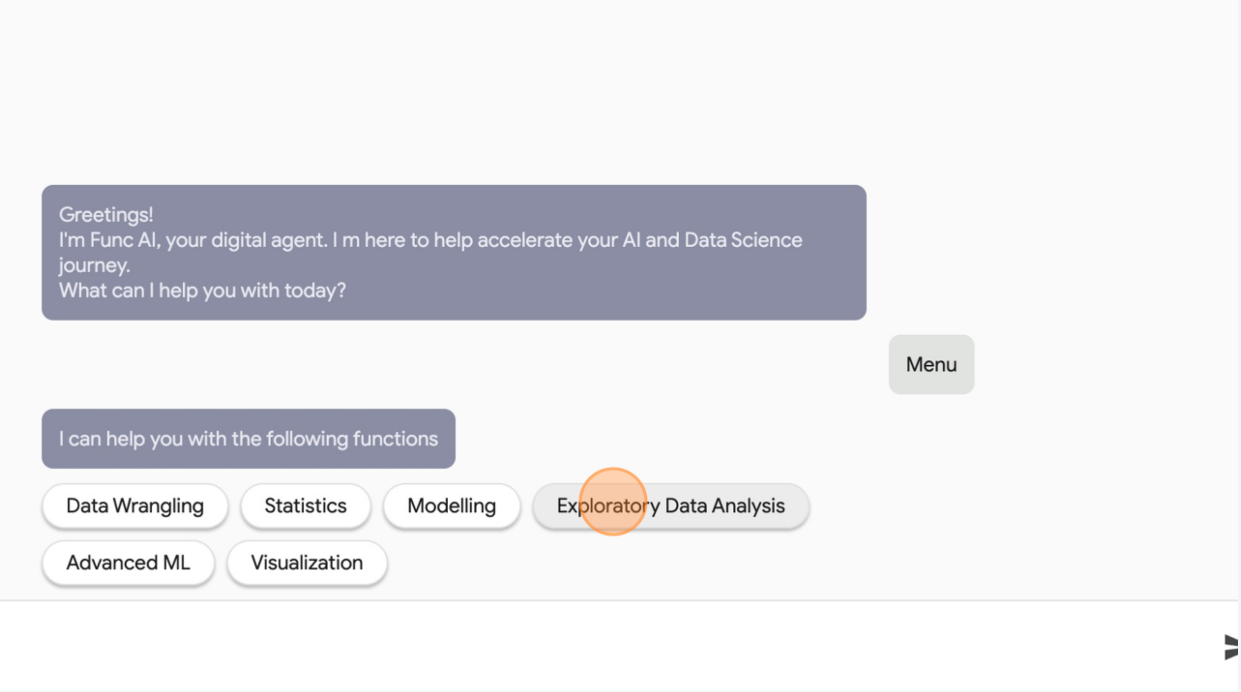


5 Begin your conversation by clicking on the Menu chip



Greetings!
I'm Func AI, your digital agent. I m here to help accelerate your AI and Da
journey.
What can I help you with today?

6 This brings the steps in a Data Science process. For the first code you will generate, select Exploratory Data Analysis

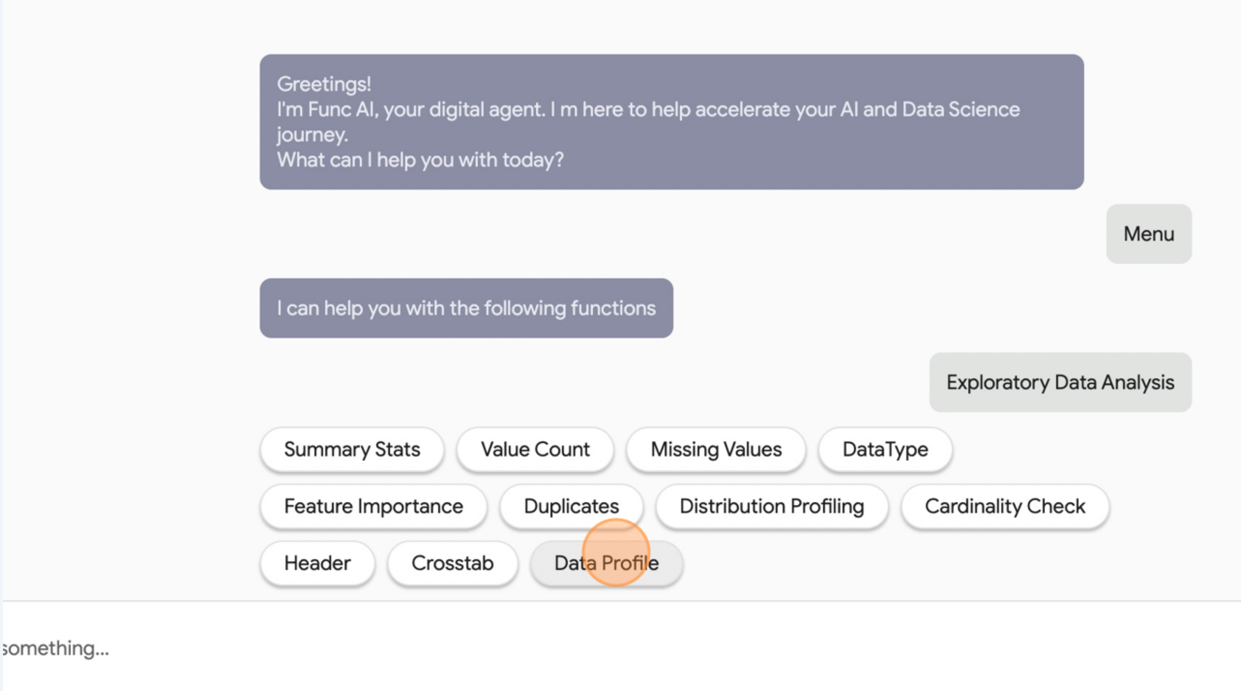


Greetings!
I'm Func AI, your digital agent. I m here to help accelerate your AI and Data Science journey.
What can I help you with today?

Menu

I can help you with the following functions

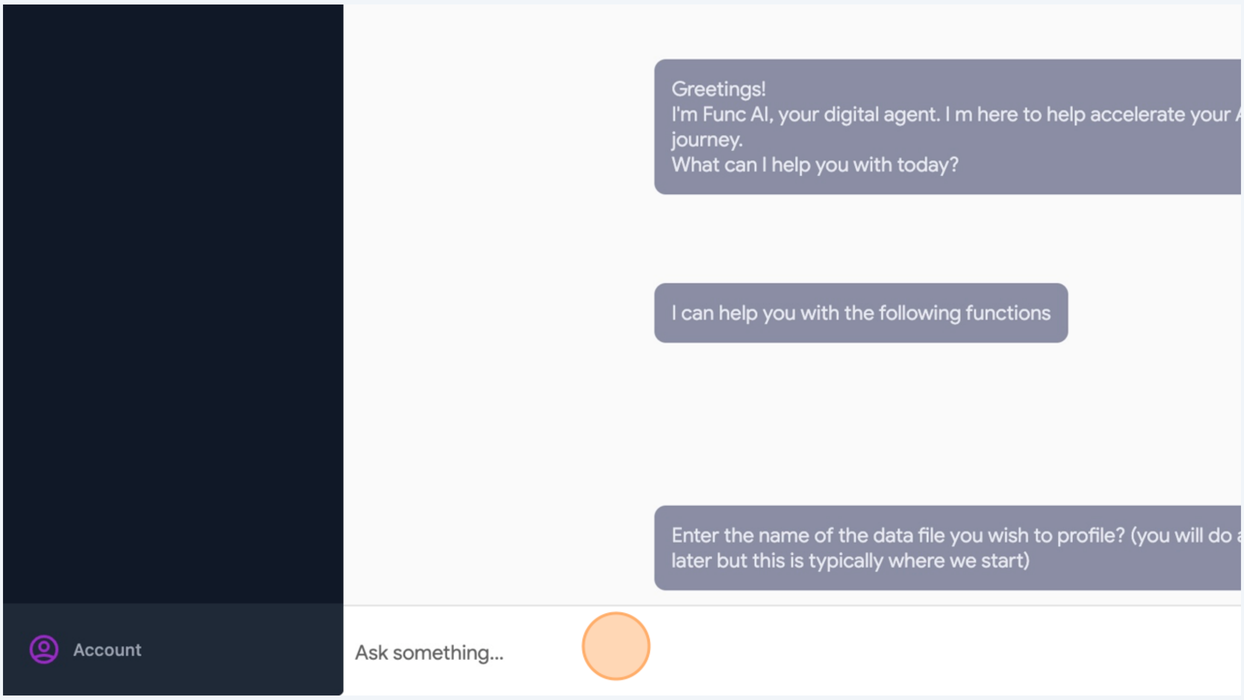
7 Now select the Data Profile chip



Greetings!
I'm Func AI, your digital agent. I m here to help accelerate your AI and Data Science journey.
What can I help you with today?

I can help you with the following functions

8 This will prompt you for responses that will help generate the correct Python code.



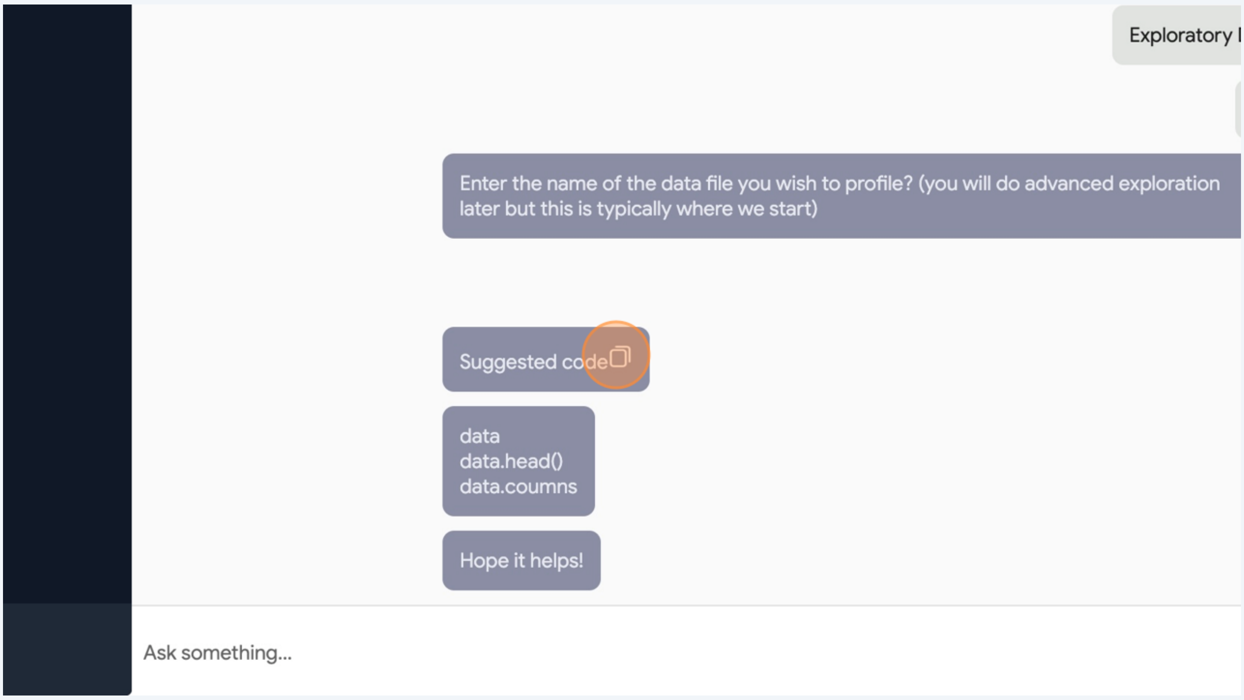
Greetings!
I'm Func AI, your digital agent. I m here to help accelerate your A
journey.
What can I help you with today?

I can help you with the following functions

Enter the name of the data file you wish to profile? (you will do a
later but this is typically where we start)

9 Type "data **Enter**"

10 This now generates optimised Python code. Click on the Copy icon to copy the code.



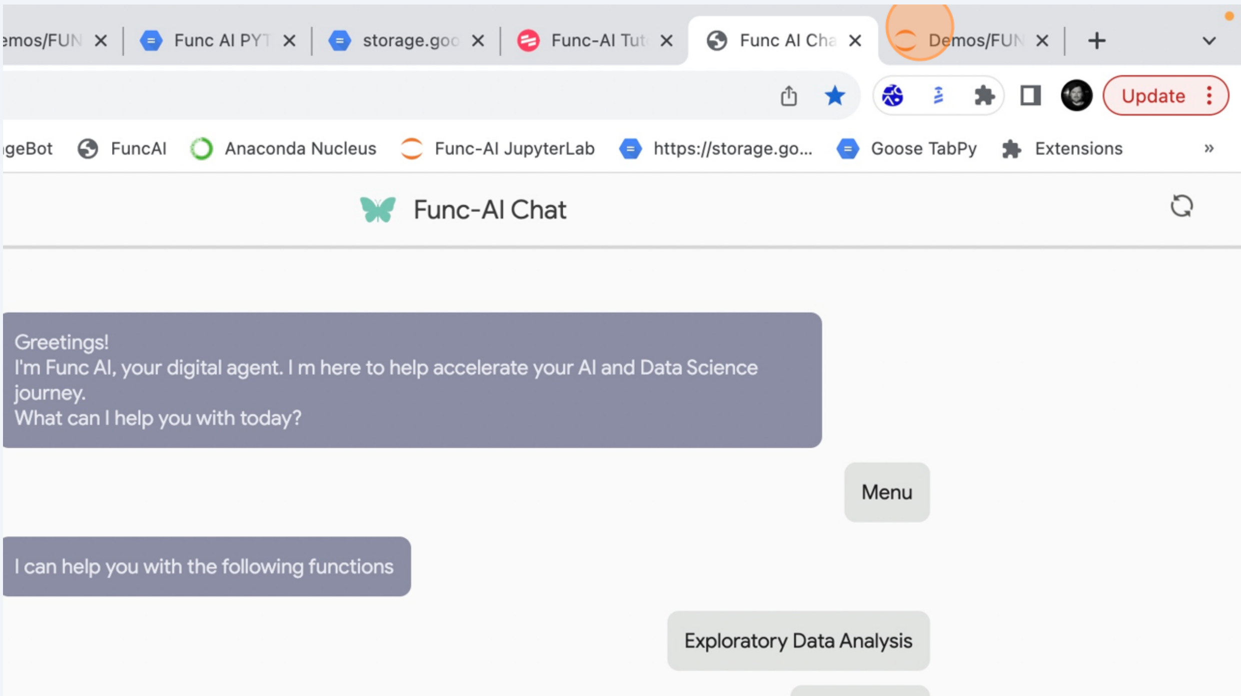
Enter the name of the data file you wish to profile? (you will do advanced exploration later but this is typically where we start)

Suggested code

```
data
data.head()
data.coumns
```

Hope it helps!

11 Return to your Python Notebook.



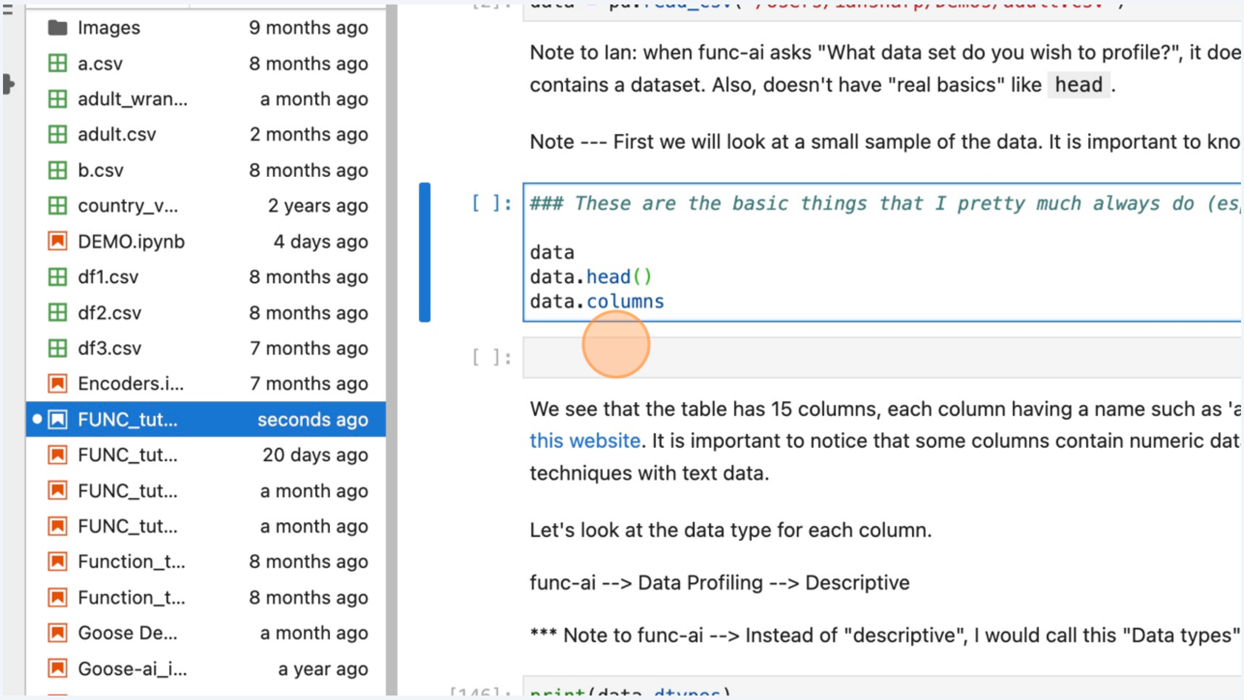
Greetings!
I'm Func AI, your digital agent. I m here to help accelerate your AI and Data Science journey.
What can I help you with today?

Menu

I can help you with the following functions

Exploratory Data Analysis

12 Paste the code into the blank cell



Images	9 months ago
a.csv	8 months ago
adult_wran...	a month ago
adult.csv	2 months ago
b.csv	8 months ago
country_v...	2 years ago
DEMO.ipynb	4 days ago
df1.csv	8 months ago
df2.csv	8 months ago
df3.csv	7 months ago
Encoders.i...	7 months ago
FUNC_tut...	seconds ago
FUNC_tut...	20 days ago
FUNC_tut...	a month ago
FUNC_tut...	a month ago
Function_t...	8 months ago
Function_t...	8 months ago
Goose De...	a month ago
Goose-ai_i...	a year ago

Note to Ian: when func-ai asks "What data set do you wish to profile?", it doe contains a dataset. Also, doesn't have "real basics" like `head`.

Note --- First we will look at a small sample of the data. It is important to kno

```
[ ]: ### These are the basic things that I pretty much always do (es

     data
     data.head()
     data.columns
```

```
[ ]:
```

We see that the table has 15 columns, each column having a name such as 'a this website. It is important to notice that some columns contain numeric dat techniques with text data.

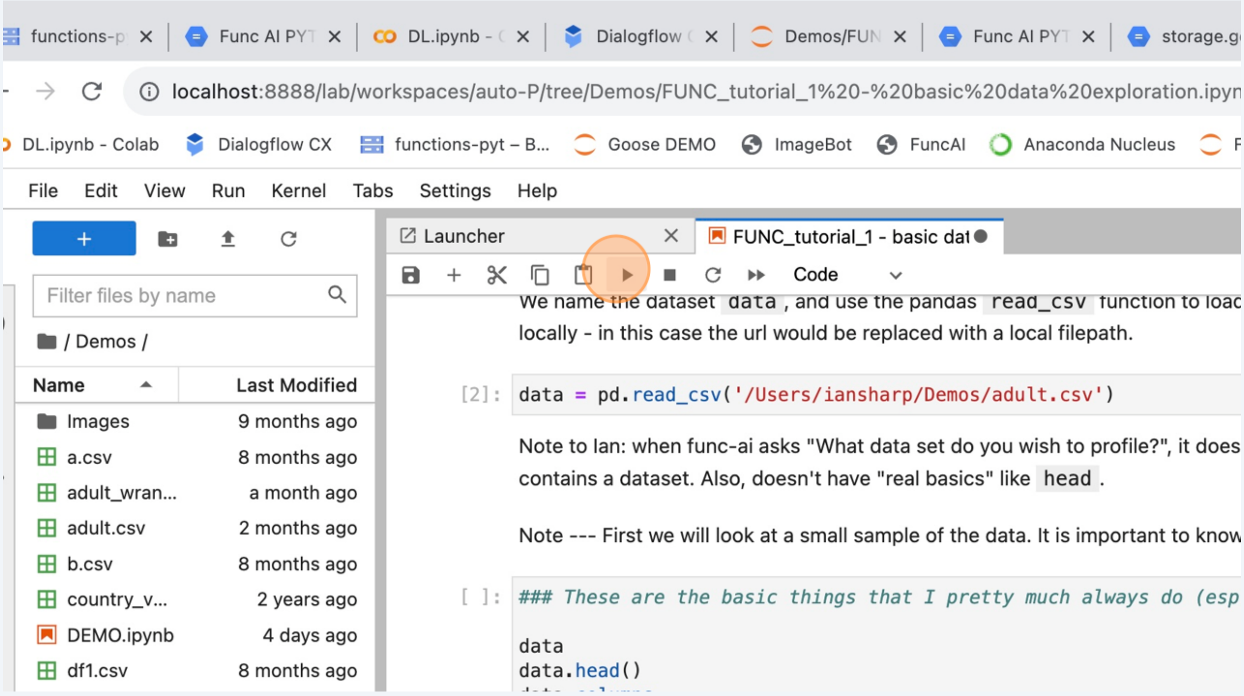
Let's look at the data type for each column.

func-ai --> Data Profiling --> Descriptive

*** Note to func-ai --> Instead of "descriptive", I would call this "Data types"

```
[146]: print(data.dtypes)
```

13 Press the Play button to execute the code.



Name	Last Modified
Images	9 months ago
a.csv	8 months ago
adult_wran...	a month ago
adult.csv	2 months ago
b.csv	8 months ago
country_v...	2 years ago
DEMO.ipynb	4 days ago
df1.csv	8 months ago

We name the dataset `data`, and use the pandas `read_csv` function to loac
locally - in this case the url would be replaced with a local filepath.

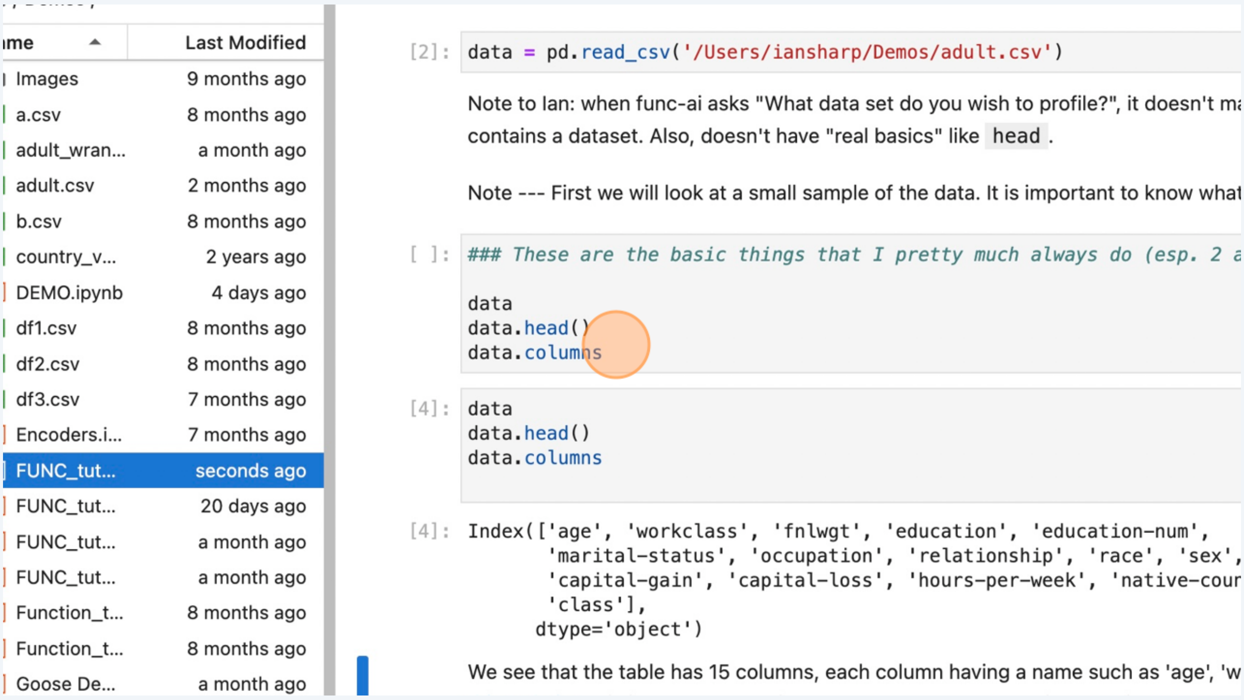
```
[2]: data = pd.read_csv('/Users/iansharp/Demos/adult.csv')
```

Note to Ian: when func-ai asks "What data set do you wish to profile?", it does
contains a dataset. Also, doesn't have "real basics" like `head`.

Note --- First we will look at a small sample of the data. It is important to know

```
[ ]: ### These are the basic things that I pretty much always do (esp

     data
     data.head()
```

14 This now executes the code and your should see the results of this basic data profile in the cell below. Congratulations!! You've just successfully run your first Python analysis. The journey has begun :)



ıme	Last Modified
Images	9 months ago
a.csv	8 months ago
adult_wran...	a month ago
adult.csv	2 months ago
b.csv	8 months ago
country_v...	2 years ago
DEMO.ipynb	4 days ago
df1.csv	8 months ago
df2.csv	8 months ago
df3.csv	7 months ago
Encoders.i...	7 months ago
FUNC_tut...	seconds ago
FUNC_tut...	20 days ago
FUNC_tut...	a month ago
FUNC_tut...	a month ago
Function_t...	8 months ago
Function_t...	8 months ago
Goose De...	a month ago

```
[2]: data = pd.read_csv('/Users/iansharp/Demos/adult.csv')
```

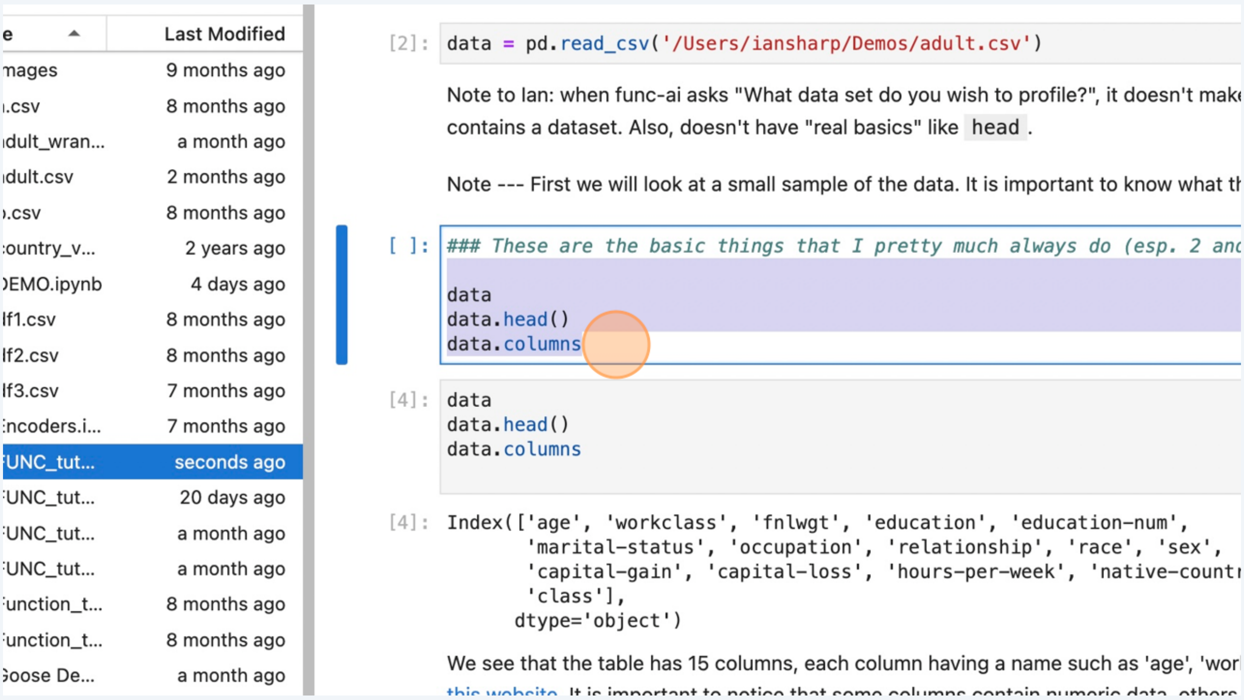
Note to Ian: when func-ai asks "What data set do you wish to profile?", it doesn't ma
contains a dataset. Also, doesn't have "real basics" like `head`.

Note --- First we will look at a small sample of the data. It is important to know what

```
[ ]: ### These are the basic things that I pretty much always do (esp. 2 a

     data
     data.head()
     data.columns

[4]: data
     data.head()
     data.columns

[4]: Index(['age', 'workclass', 'fnlwgt', 'education', 'education-num',
            'marital-status', 'occupation', 'relationship', 'race', 'sex',
            'capital-gain', 'capital-loss', 'hours-per-week', 'native-cour
            'class'],
           dtype='object')
```

We see that the table has 15 columns, each column having a name such as 'age', 'w

15 Click here



e	Last Modified
mages	9 months ago
.csv	8 months ago
dult_wran...	a month ago
dult.csv	2 months ago
.csv	8 months ago
ountry_v...	2 years ago
DEMO.ipynb	4 days ago
lf1.csv	8 months ago
lf2.csv	8 months ago
lf3.csv	7 months ago
ncoders.i...	7 months ago
UNC_tut...	seconds ago
UNC_tut...	20 days ago
UNC_tut...	a month ago
UNC_tut...	a month ago
unction_t...	8 months ago
unction_t...	8 months ago
Goose De...	a month ago

```
[2]: data = pd.read_csv('/Users/iansharp/Demos/adult.csv')
```

Note to Ian: when func-ai asks "What data set do you wish to profile?", it doesn't make
contains a dataset. Also, doesn't have "real basics" like head.

Note --- First we will look at a small sample of the data. It is important to know what th

```
[ ]: ### These are the basic things that I pretty much always do (esp. 2 and

     data
     data.head()
     data.columns
```

```
[4]: data
     data.head()
     data.columns
```

```
[4]: Index(['age', 'workclass', 'fnlwgt', 'education', 'education-num',
            'marital-status', 'occupation', 'relationship', 'race', 'sex',
            'capital-gain', 'capital-loss', 'hours-per-week', 'native-countr
            'class'],
           dtype='object')
```

We see that the table has 15 columns, each column having a name such as 'age', 'wor

this website. It is important to notice that some columns contain numeric data, others

16 For subsequent steps in the tutorial, simply use Func-AI in the same way but follow the inline instructions within your Tutorial Jupyter Notebooks.

An example would be:

funcai-ai --> Menu --> Exploratory Data Analysis --> Data Type

Tutorial 1: Basic Exploratory Data Analysis

The following steps represent the step by step selections and entries for the tutorial. Python is case sensitive so it is important to enter the fields precisely as specified. See Step 16 for more explanation

17 Exploratory Data Analysis -> Header -> data

18. Exploratory Data Analysis -> Data Type -> data

19. Exploratory Data Analysis -> Summary Stats -> data -> Cross Sectional

Tip! Rather than go through the Exploratory Data Analysis menu, you can search on keywords to fast track the conversation. For example, in the next function, simply type Feature Profiles

20. Feature Profiles -> data -> marital-status

21. Value Count -> data -> marital-status, sex

22. Cardinality Check -> data

23. Menu -> Visualisation -> Heatmap -> data -> marital-status -> workclass -> hours-per-week -> mean

24. Histogram -> data -> fnlwgt

Tutorial 2: Fitting a Simple Model

25 Data Type -> data

26 Feature Importance -> data -> class

27 Menu -> Modelling ->. Random Forests -> data -> class -> capital-gain, capital-loss -> Select 1 (the speed option) -> 0.3 Yes (to Visualisation prompt)

28 Menu -> Data Wrangling -> PIPELINES (Pre Encoder) -> data -> age, fnlwgt, capital-gain, capital-loss, hours-per-week -> marital-status, occupation, sex, native-country, workclass -> Select 'No' to Ordinal Data prompt

Tutorial 3: Data Wrangling

29 Recode -> data -> workclass -> Many To One -> public-sector -> Local-gov, State-gov, Federal-gov

30 Filter -> data -> workclass == 'public-sector'

31 Sampling -> data -> Stratified -> workclass -> Proportionate -> 0.1

32 Value Count -> data -> workclass

33 Missing Values Treatment -> data -> Fill -> 'No' to Time Series -> age -> 75

34 Bin -> data -> age -> Quantiles

ⓘ To save your work in this Notebook, click on the Save icon in the top left hand corner of it. This will save your Notebook content and create a checkpoint

Tutorial 4: Unsupervised Models

35 Menu -> Modelling -> Clusters -> data -> Yes -> age, hours-per-week -> age -> hours-per-week -> 2 -> K Means Clustering

36 Boxplot -> data ->age, fnlwgt, education-num, capital-gain, capital-loss, hours-per-week -> 'No' to Filtering values

37 Normalise -> data

38 Clusters -> scaled_df -> No -> age -> hours-per-week -> 2 -> K Means Clustering

Note: this step takes the scaled data generated in Step 37 as the input. This is useful as it introduces how we connect steps in our Python analysis. The previous step has also created our data as a Pandas dataframe (as oppose to a numpy array) which means it works well with sci-kit learn libraries and other modelling techniques.

39 Dimensionality Reduction -> data -> age, fnlwgt, education-num, capital-gain, capital-loss, hours-per-week -> 3 -> No

40 Isolation Forests -> data -> age, fnlwgt, capital-gain, capital-loss, hours-per-week -> 0.3 -> select Yes (to visualisation prompt)