

IBM watsonx.ai

Technical Hands-on Lab
New models and
Synthetic data

Felix Lee
felix@ca.ibm.com
Worldwide Technology Enablement

Anshupriya Srivastava
Anshupriya.Srivastava@ibm.com
Worldwide Technology Enablement



Contents

1. Introducing watsonx.ai	3
2. About this Lab.....	5
2.1 Disclaimer	5
3. Getting Help.....	7
4. Prerequisites & Getting Started	7
4.1 Obtain an IBM Cloud Account.....	7
5. Foundation models update	8
5.1 IBM Granite models	8
5.1.1 Advantages of IBM models	10
6. Synthetic data generation.....	11
6.1 Create synthetic data by defining customized data schema.....	16
6.1.1 Using watsonx.ai to examine and update your synthetic data.....	28
6.2 Create synthetic data by mimicking existing data/schema	36
6.2.1 Adding data asset to your sandbox project.....	36
6.2.2 Generating data with a seed file	37
Appendix A. Revision History.....	46

1. Introducing watsonx.ai

Watsonx.ai is a core component of Watsonx, IBM's enterprise-ready AI and data platform designed to multiply the impact of AI across an enterprise.

The Watsonx platform has three powerful components:

- **watsonx.ai** studio for new foundation models, generative AI and Machine Learning (traditional AI)
- **watsonx.data** fit-for-purpose data store that provides the flexibility of a data lake with the performance of a data warehouse
- **watsonx.governance** toolkit, which enables AI workflows that are built with responsibility, transparency, and explainability.

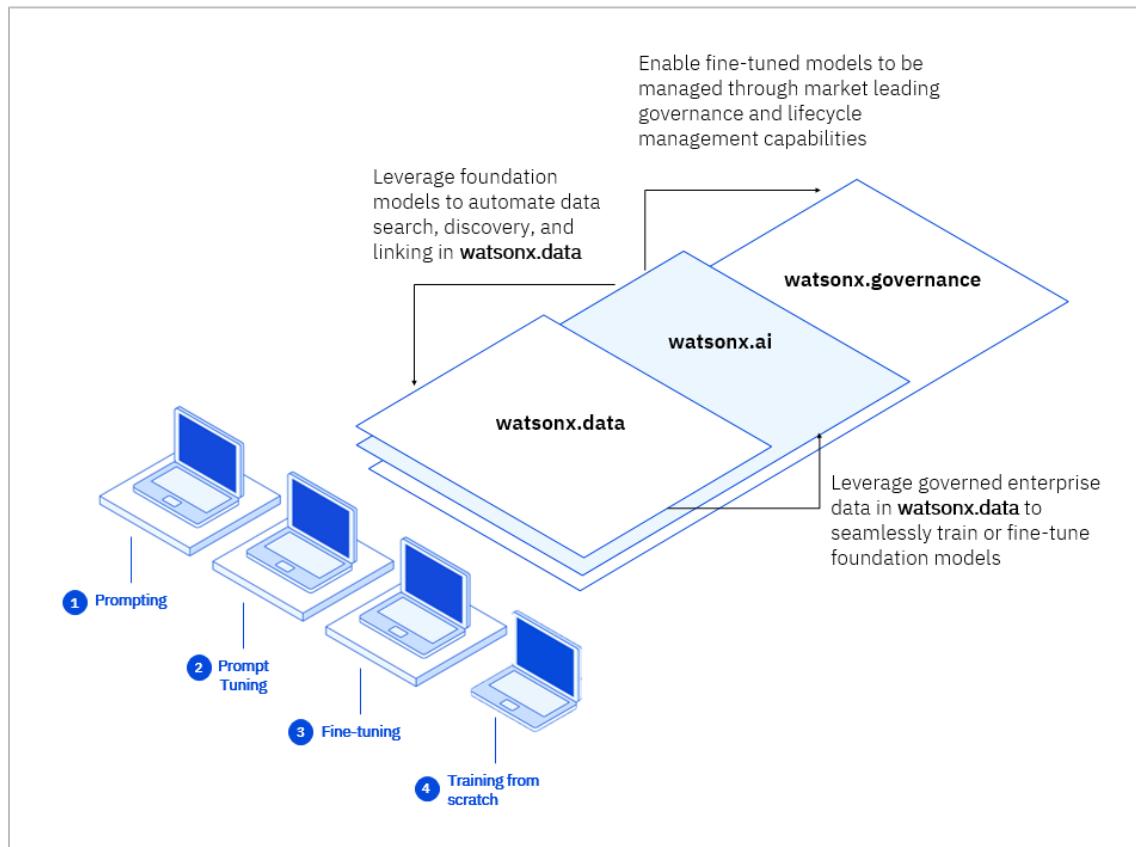


Figure 1: IBM Watsonx platform

The Watsonx.ai component (the focus of this lab) makes it possible for enterprises to train, validate, tune, and deploy AI models – both traditional AI and generative AI. With Watsonx.ai, enterprises can leverage their existing traditional AI investments as well as exploit the innovations and the potential of generative AI using foundation models to bring advanced

automation and AI-infused applications to reduce cost, improve efficiency, scale, and accelerate the impact of AI across their organizations.

2. About this Lab

This IBM watsonx.ai Technical Hands-on Lab Part II is a supplement to the [IBM watsonx.ai Technical Hands-on Lab Part 1](#). It is assumed that you have completed this first Hands-on lab. Detailed instructions and screen flow available there are not repeated here.

In this lab, you will be introduced to the following watsonx.ai capabilities:

- New foundation models available since the publication of the original [watsonx.ai Technical Sales Lab Guide Part 1](#).
- Generating synthetic data for use with traditional AI or foundation models testing/tuning. This lab details two ways to generate synthetic data:
 - **Generate:** Generate different data sets either from scratch
 - **Mimic:** Generate new data based on a small existing data set

You can use the same procedure from [Part 1 of the lab](#) to set up an environment to work with the labs here.

2.1 Disclaimer

IBM watsonx.ai is developed and released in an agile manner. In addition to constantly adding new capabilities, the web interface is likely to change over time. Therefore, the screenshots used in this lab may not always look exactly like what you see. You can expect to encounter some of the following:

- Additional foundation models in the library list
- Additional foundation models available for prompt tuning
- Changes in the user interface (location of buttons, text for various fields)
- Additional tabs/buttons

These should not affect how the labs work, but have patience and be prepared to explore a little bit until this lab gets updated.

There are three changes, however, that can affect the results. For example:

- Foundation models can be very sensitive to input. If you enter slightly different text than what the exercise is using (even if it is just one single word or a modified set of labeled data), the outcome can be very different.
- There is ongoing tuning of the models. If the models themselves are updated, then some of the results may vary.

Please post any questions on the [#data-ai-demo-feedback](#) Slack channel (IBMer only). IBM partners can request help at the [Partner Plus Support](#) website.

3. Getting Help

Lab guide help: If you require assistance in interpreting any of the steps in this lab, please post your questions to the [#data-ai-demo-feedback](#) Slack channel (IBMer only). Business Partners can request help at the [Partner Plus Support](#) website.

Troubleshooting: See the [TechZone Set Up Troubleshooting Guide](#) for a list of the common issues and solutions/workarounds when using TechZone.

IBM watsonx.ai: Assistance with the watsonx.ai product itself is available in the [#watsonx-ai-feedback](#) (IBMer) and [#watsonx-ai-enablement](#) Slack channels (IBMer only). Additionally, please refer to the [watsonx.ai documentation](#) as needed.

4. Prerequisites & Getting Started

You will need an IBM Cloud account to gain access to the TechZone account that hosts the various Watson and watsonx services used in this lab.

4.1 Obtain an IBM Cloud Account

If you have an IBM Cloud account, you can skip this step. If you do not have an IBM Cloud account, [Click this link](#) to create one. After registration, you will be sent an email to activate your account. This can take a few hours to process. Once you receive the confirmation email, follow the instructions provided in the email to activate your account.

You can use **IBM TechZone** to perform the exercises in this lab. The detailed setup instructions are available in [this document](#) (Section 4 of the L3 Lab Part 1). You must complete the steps before you can proceed with any of the exercises in this lab.

It is assumed that you have completed [Part 1](#) of the L3 lab (or are familiar with the concepts and contents).

5. Foundation models update

IBM continues to roll out new foundation models in watsonx.ai. You should go to [Supported foundation models available with watsonx.ai](#) to find the latest list. As of January 2024, the list is as follows:

IBM models:

- [granite-13b-chat-v1](#)
- [granite-13b-chat-v2](#)
- [granite-13b-instruct-v1](#)
- [granite-13b-instruct-v2](#)

Models provided by Hugging Face:

- [flan-t5-xl-3b](#)
- flan-t5-xxl-11b
- flan-ul2-20b
- gpt-neox-20b
- [llama-2-13b-chat](#)
- llama-2-70b-chat
- mpt-7b-instruct2
- mt0-xxl-13b
- starcoder-15.5b

The 6 highlighted models were introduced since the initial release of [Part 1](#) of the Technical Hands-on guide.

The llama-2-13b-chat model is a smaller version of llama-2-70b-chat. Likewise, the flan-t5-xl-3b model is a smaller version of flan-t5-xxl-11b, and at 3 billion (3b) is currently the smallest model on the watsonx.ai supported model list. You can try these out and observe some advantages of smaller models. These include:

- More cost efficient
- Likely less verbose and creative – but completion still heavily depend on the input prompt and any inference parameter tuning

5.1 IBM Granite models

IBM is rolling out its proprietary models. These are built on highly curated data sets that have been vigorously filtered to remove:

- Hate, Abuse, and Profanity content
- Copyright and licensed materials
- Duplications
- Any other undesirable, blacklisted material, and blocked URLs

IBM offers Indemnification for its models (expand and read the Intellectual Property Protection for AI Models section in [Foundation models in Watsonx.ai](#)). Clients can use the Granite (and future IBM models like Obsidian which is coming in 2024) with confidence.

Here are the descriptions of the two Granite variants:

- **granite-13b-instruct-v1**

Instruct models are tuned to work well with input instructions, typically entered as part of the prompt. This variant is a Supervised Fine-Tuned (SFT) version of the base model to improve its instruction-following. It was tuned using a mix of FLAN and a mixture of other datasets (Dolly, HHRLHF, and IBM internal datasets, etc.). This model is intended as a starting point to help bootstrap further downstream alignment or task-specific tuning.

- **granite-13b-chat-v1**

This variant is a purpose-aligned version of the granite.13b.instruct variant specifically for chats. It was aligned using Contrastive Fine Tuning (CFT) to improve its harmlessness and the quality of its generation responses. This model should be used when looking to prompt-engineer out of the box, particularly when longer responses are desired. It also may be helpful as a starting point for further downstream fine-tuning.

In Dec 2023, IBM added version 2 variants of the 2 Granite models: granite-13b-chat-v2 and granite-13b-instruct-v2.

The focus of this part of the lab is not on prompt engineering. You can retry some of the prompt engineering exercises from [Part 1](#) and use the Granite model instead. In this lab's Prompt Tuning section, you will try out the flan-t5-xl-3b model (the Granite model will be available for prompt tuning later in 2024).

The version 1 Granite models were trained on 1 trillion tokens, whereas the version 2 models were trained on 2.5 trillion tokens. The version 2 models are much more efficient and they perform better than the version 1 models. In Proof of Experience (PoX), sellers should get familiar with the v2 models (or the latest version of the model) and use them.

5.1.1 Advantages of IBM models

There are many reasons why clients should consider IBM's foundation models; some of which include:

- IBM models are built on highly curated, filtered, and refined data. IBM removes the following contents before using the data to build IBM's Large Language Models (LLMs)
 - Hate, Abuse, and Profanity content
 - Copyright and licensed materials
 - Duplications
 - Any other undesirable, blacklisted material, and blocked URLs

As a result, clients can use IBM models with confidence and trust.

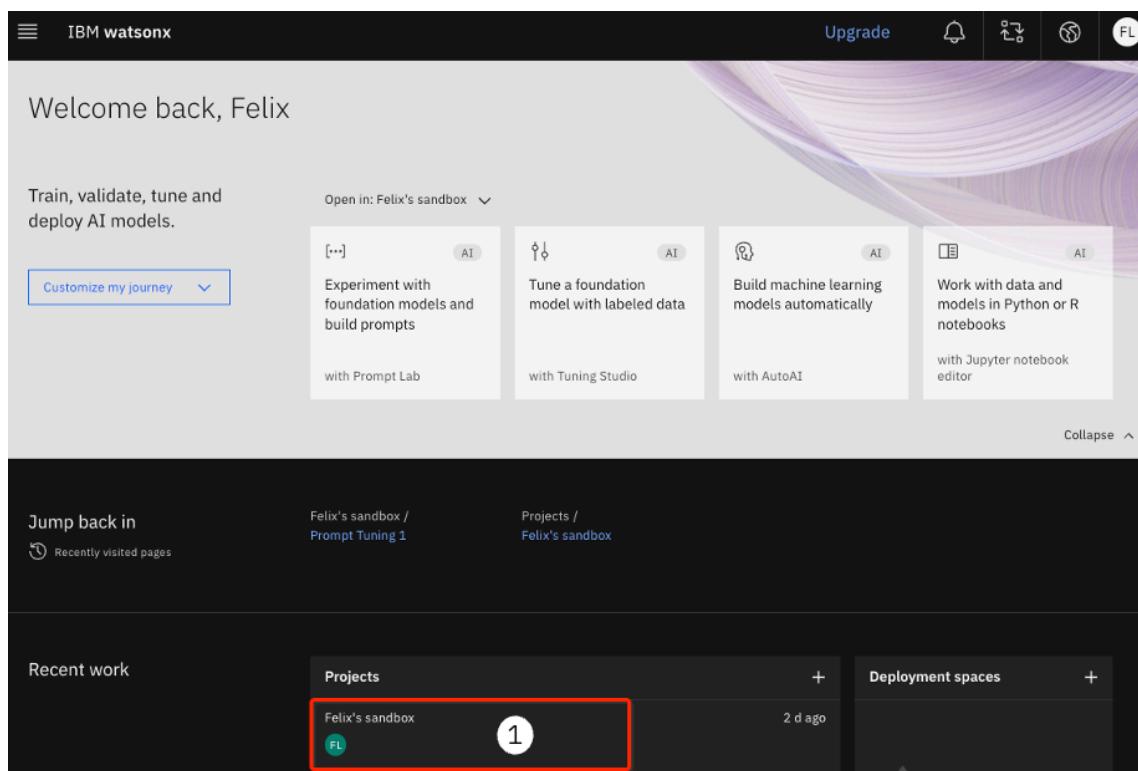
- In addition, since the data used to build the model is completely known, IBM models provide complete transparency and explainability. This is typically not available for other models.
- Because of the vigorous cleaning and filtering, IBM models may be smaller by comparison, but they are targeted for business applications. Clients incur lower costs but still reap similar benefits when compared with larger Large Language Models (LLMs) such as GPT, and with data explainability.
- IBM provides indemnification for using IBM models (see [this](#) for more details).

6. Synthetic data generation

Synthetic data is data that is artificially generated using advanced statistics, as opposed to real data, which is gathered by observing real-world events. Synthetic data can be used with [watsonx.ai](#) to augment or replace real data for improving AI models, protecting sensitive data, and mitigating bias. Synthetic data can be generated using foundation models to replicate data with the right statistical attributes. Synthetic data can be used for hackathons, product demos, internal prototyping, exploring market behaviors, or a large number of other use cases that require more data or have privacy concerns.

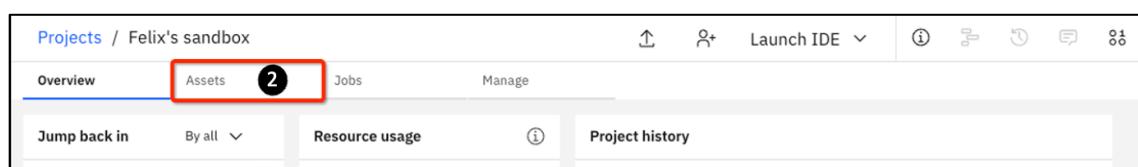
In this section, you will take a break from investigating foundation models to look at synthetic data generation, another important feature of watsonx.ai.

1. Open the watsonx.ai Prompt Lab and click your project. This is most likely called <your username> sandbox.



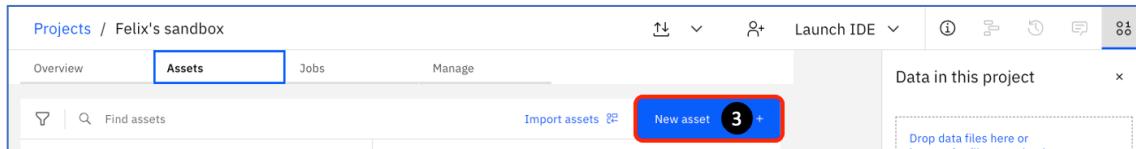
The screenshot shows the IBM WatsonX interface. At the top, it says "Welcome back, Felix". Below that, there's a section titled "Train, validate, tune and deploy AI models." with a "Customize my journey" button. To the right, there are four cards: "Experiment with foundation models and build prompts with Prompt Lab", "Tune a foundation model with labeled data with Tuning Studio", "Build machine learning models automatically with AutoAI", and "Work with data and models in Python or R notebooks with Jupyter notebook editor". Below this, there's a "Jump back in" section with a "Recently visited pages" link. The main area shows "Recent work" with a card for "Felix's sandbox / Prompt Tuning 1" from "2 d ago". At the bottom, there are tabs for "Projects" and "Deployment spaces", and a navigation bar with "Overview", "Assets" (highlighted with a red box), "Jobs", "Manage", and "Resource usage", "Project history", and "Launch IDE".

2. Select the **Assets** tab.

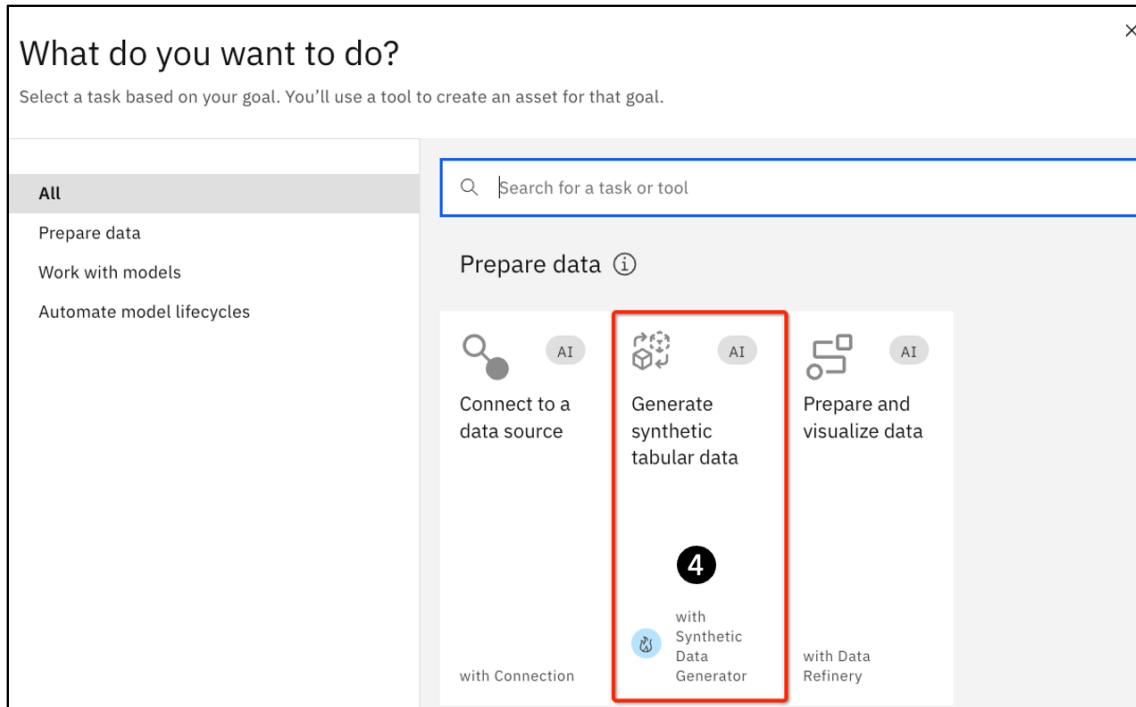


The screenshot shows the "Assets" tab selected in the navigation bar. Below it, there are three tabs: "Jump back in", "Resource usage", and "Project history".

3. Click **New asset**.



4. Click on the **Generate synthetic tabular data** tile.



5. On the next panel, provide a name such as **Data Generate Test 1**. You can also optionally provide a description in the **Description** field
6. Click **Create**.

Generate synthetic tabular data

Define the details to create a synthetic data flow asset and open it in the Synthetic Data Generator tool.

+ New

Samples

Local file

Define details

Name

Data Generate Test 1 **5**

Description (Optional) 0/500

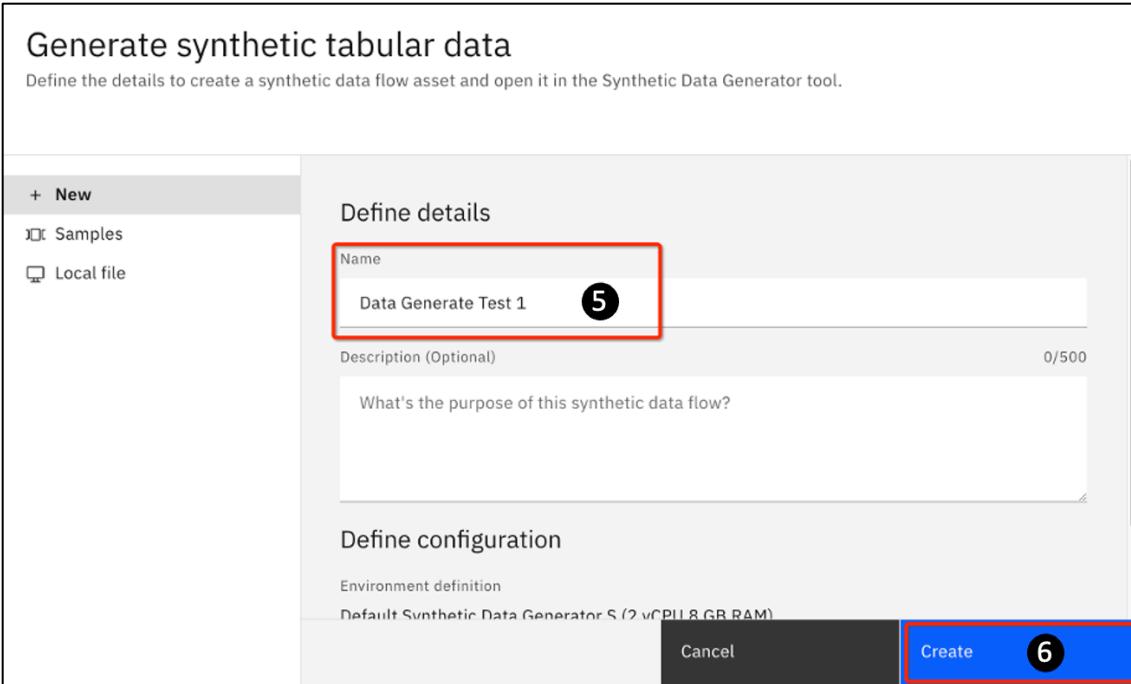
What's the purpose of this synthetic data flow?

Define configuration

Environment definition

Default Synthetic Data Generator S (2 vCPU 8 GB RAM)

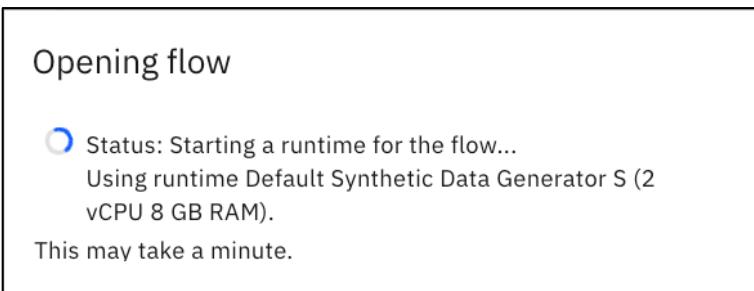
Cancel **Create 6**



You will see the following message:

Opening flow

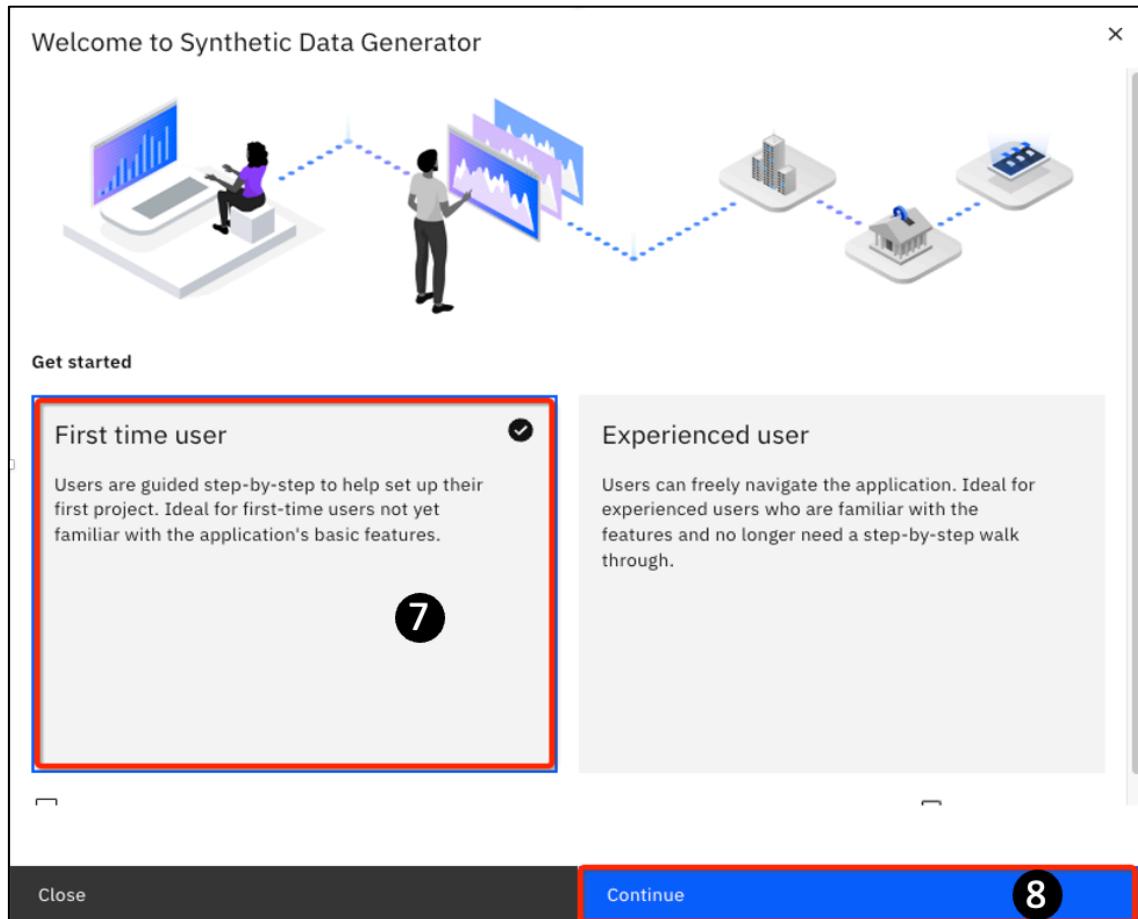
⌚ Status: Starting a runtime for the flow...
Using runtime Default Synthetic Data Generator S (2 vCPU 8 GB RAM).
This may take a minute.



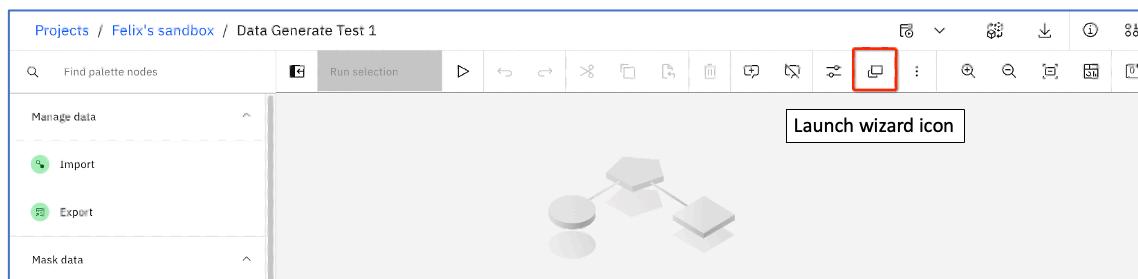
As noted in the message, this process may take a few minutes.

Note: sometimes the process may fail. If so, retry by clicking **Create** again.

7. On the subsequent panel, click on the **First time user** tile.
8. Click **Continue**.



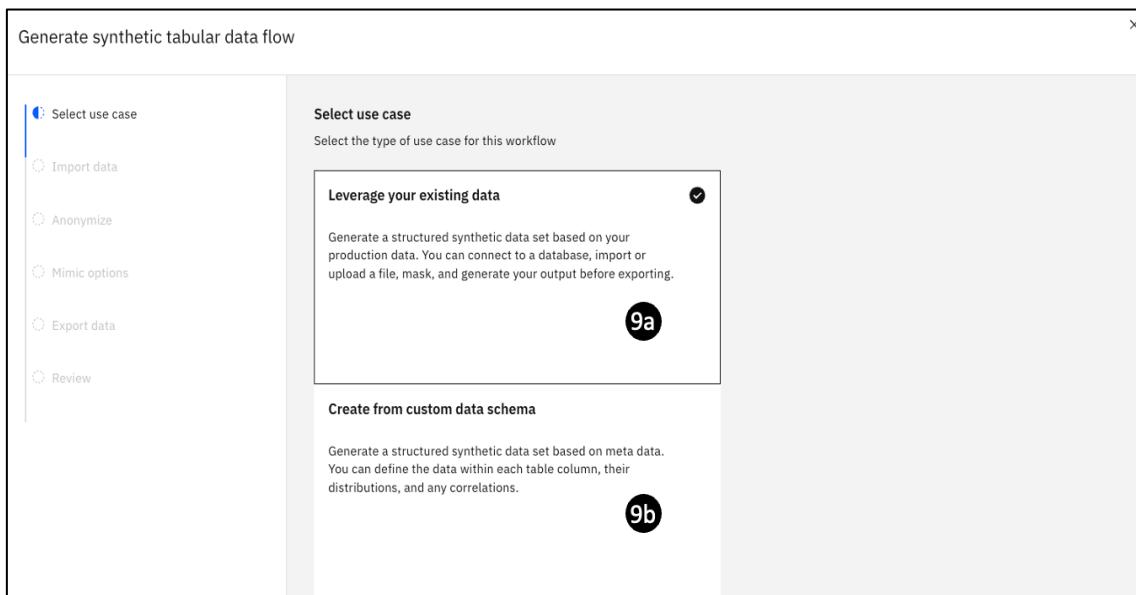
If you click on the **Experienced user** tile and then click **Continue**, you will be taken to the Synthetic data generation canvas tool for data generation. You can always come back to the **Welcome to Synthetic Data Generator** wizard by clicking on the **Launch wizard icon** (you may need to scroll or expand your window to see this icon)



9. If you select the **First-time user** tile, you will see a panel showing 2 tiled options:
- Leverage your existing data** – Use this to leverage existing data sets and generate new synthetic data that resembles (but does not include) your existing data. This is an easy way to generate data that is compatible with a specific schema.
 - Create from custom data schema** – Use this to create your own table schema and populate it with synthetic data.

You will try both methods in this lab.

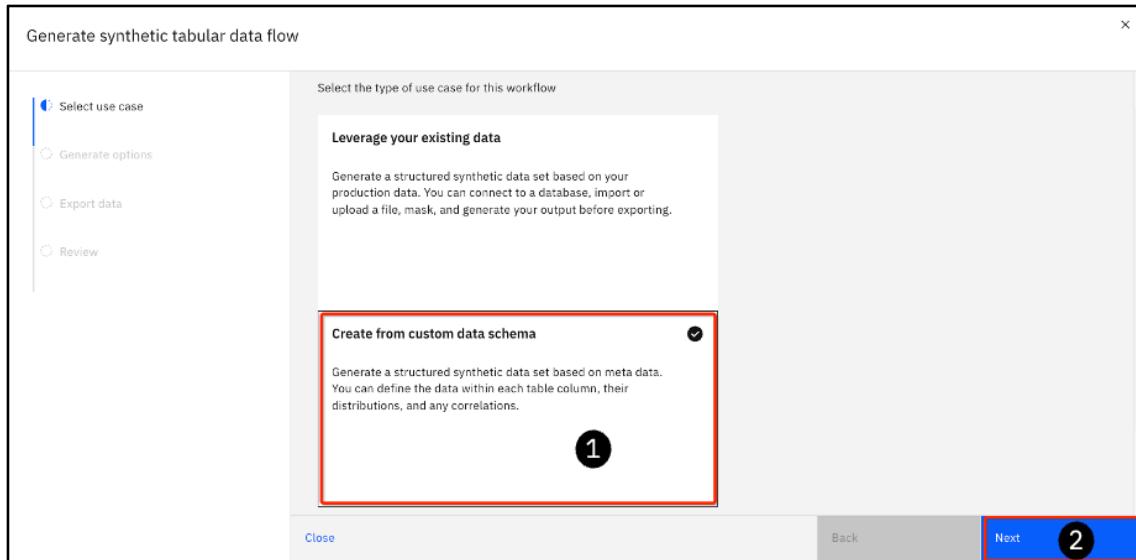
Don't select any options yet.



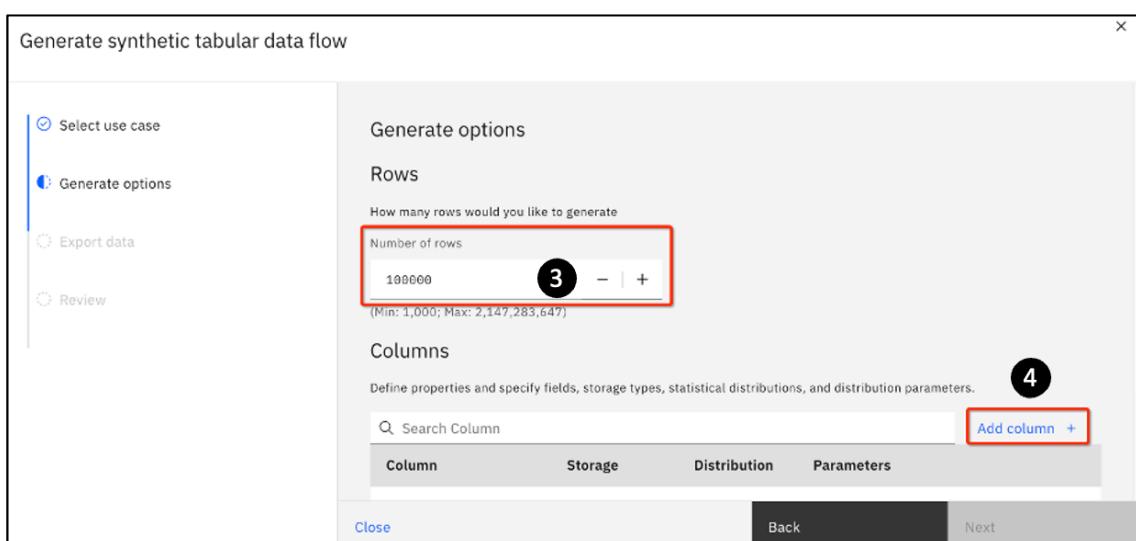
6.1 Create synthetic data by defining customized data schema

In this section, you will create synthetic data from scratch.

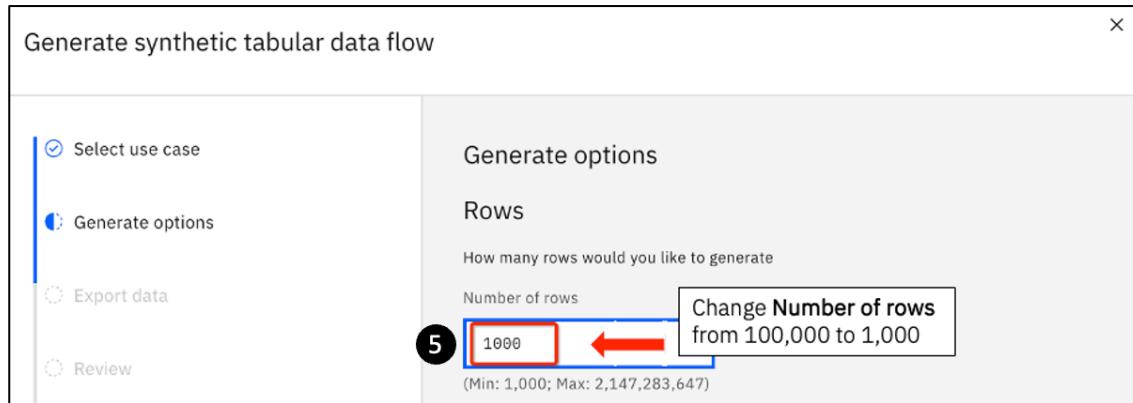
1. From the last step in Section 5, click on the **Create from custom data schema** tile.
2. Click Next.



3. The **Generate synthetic tabular data flow** page opens. You can set the number of rows to generate – the default is **100,000** rows.
4. You can also define the columns (the schema) of the table.

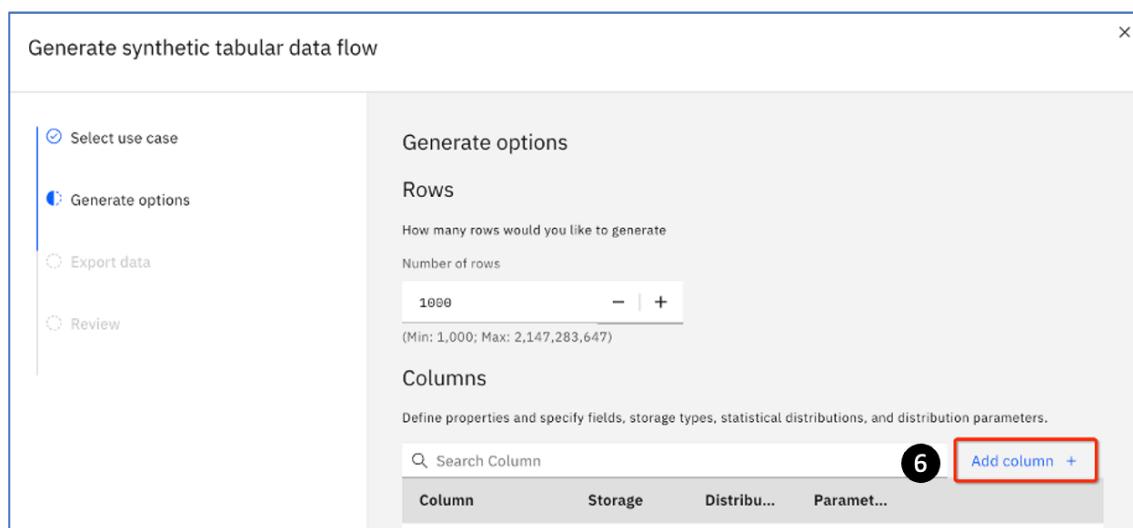


5. By default, watsonx.ai will generate 100,000 rows. For this exercise, change the **Number of rows** field to the minimum number allowed: 1,000.



6. Now you will start to define columns. You will define a simple table with an employee's last name, age, and salary range.

On this panel, click **Add column +**.



The following panel appears. Fill in the first 3 fields as follows:

7. **Column:** Enter Last_Name
8. **Storage:** Select String from the drop-down list
9. **Distribution:** Select Categorical from the drop-down list

Specify Parameters

Last_Name	7
Storage (required)	
String	8
Distribution	
Categorical	9

Since you are defining a string column, the **Distribution** is simply a categorical list of last names. In this case, watsonx.ai is asking you to provide a list of possible outcomes (here is a list of last names, in general, it would be whatever valid values that a string variable can take on). You will need to provide a list and the probability value for each item on that list; the sum must be 1.0.

When you scroll further down, you will see fields for **Value** and **Probability**. For the first name of the category, enter the following values:

10. For **Value**: Enter **Brown**

11. For **Probability**: Enter **0.075**

Specify Parameters

Categorical ▼

Distribution parameters: (required)

[Remove](#) - [Add value](#) +

Value	Probability
Brown 10	0.075 11

All probability values must sum to 1.

Use these options to reject generated values which are not wanted even though they would be valid for

Cancel Save

12. Notice that the **Save** button is still grayed out. The reason is stated in red above: **All probability values must sum to 1.** You will need to keep adding additional values until all the probability adds up to 1.0.

Click on **Add value +**.

13. A new row will be added, which you will fill in next.

Specify Parameters

Categorical

Distribution parameters: (required)

Remove **Add value +** 12

Value	Probability
Brown	0.075
13	

There are 3 error cells.

Cancel Save

Carefully add all the values in the following table (they add up to a total of 20 last names); remember you already added the **Brown** row.

Value	Probability	Value	Probability
Brown	0.075	Leung	0.096
Gordan	0.054	Wilson	0.019
Newman	0.037	Duncan	0.022
Chen	0.082	Wong	0.085
Lord	0.023	Kapoor	0.072
Muthu	0.027	Allen	0.028
Bird	0.022	Thomas	0.088
Bentley	0.024	Roberts	0.054
James	0.018	Martinez	0.066
Abrams	0.029	Jones	0.079

Once you have a list with a probability value adding up to **1.0**, the **Save** button will turn blue; if it doesn't you have an error in the **Probability** numbers ... recheck them.

14. Click **Save** to create the column.

Specify Parameters

Value	Probability
Allen	0.028
Thomas	0.088
Roberts	0.054
Martinez	0.066
Jones	0.079

Cancel Save 14

15. You are redirected to the **Generate synthetic tabular data flow** page. Notice that the **Last_Name** column is now added.

16. Click **Add column +** to add another column.

Generate synthetic tabular data flow

Select use case
Generate options

Generate options
Review

Rows

How many rows would you like to generate

Number of rows: - +

(Min: 1,000; Max: 2,147,283,647)

Columns

Define properties and specify fields, storage types, statistical distributions, and distribution parameters.

Column	Storage	Distribu...	Paramet...
Last_Name 15	String	Categorical	{Brown=0....}

Add column + 16

Insert the following data:

17. For Column: Enter **Age**

18. For Storage: Select **Integer** from the drop-down

19. For Distribution: Select **Normal** from the drop-down

With an integer column, you can optionally use a **Categorical** distribution (like you did in Step 9). In that case, you will need to provide a list of ages and the probability of each, similar to what you did with **Last_Name**. For this lab, you will use a typical normal distribution for number-based items (this is the default assumed by watsonx.ai)

20. Notice that watsonx.ai automatically filled in a mean value of **50**.

21. Watsonx.ai also puts in a default **Stddev** (standard deviation) value of **10**.

As you are working with a global company, you might want to have a larger spread of data. Keep the mean value for this lab, but change the **Stddev** to **15** instead of **10**.

22. Click **Save**.

The screenshot shows the 'Specify Parameters' dialog box. It has sections for Column (Age), Storage (Integer), Distribution (Normal), Mean (50), and Stddev (15). The 'Save' button is highlighted with a red border.

Parameter	Value	Step Number
Column (required)	Age	17
Storage (required)	Integer	18
Distribution	Normal	19
Mean (required)	50	20
Stddev (required)	15	21
Buttons		22

23. The **Generate Synthetic tabular data flow** reopens. The **Last_Name** and **Age** columns are now added.

24. Click **Add column +** to add a third and final column.

Generate synthetic tabular data flow

Select use case

Generate options

Export data

Review

Generate options

Rows

How many rows would you like to generate

Number of rows

1000

(Min: 1,000; Max: 2,147,283,647)

Columns

Define properties and specify fields, storage types, statistical distributions, and distribution parameters.

Q Search Column

Column	Storage	Distribution	Parameters
Last_Name	String	Categorical	[Brown=0.078,Gor...
Age	Integer	Normal	[mean=50,stddev...

Add column +

Close Back Next

25. For the last column, use the following values. For **Column**: Enter **Salary**.
26. For **Storage**: Select **Real** from the drop-down
27. For **Distribution**: Select **Normal** from the drop-down
28. For **Mean**: Enter **140000**
29. For **Stddev**: Enter **35000**
30. Select the **Specify minimum** check box. Enter **80000** for the **Reject values below:** field.

In this example, you want to ensure that you generate a range where the salary cannot be lower than \$80,000. You may want to do this to eliminate a salary that is valid but not desirable.

You may also define a **maximum**. In this lab it is not set.

31. Click **Save**.

Specify Parameters

Column (required)

Salary **25**

Storage (required)

Real **26**

Distribution

Normal **27**

Mean (required)

140000 **28**

Stddev (required)

35000 **29**

Use these options to reject generated values which are not wanted even though they would be valid for the specified distribution.

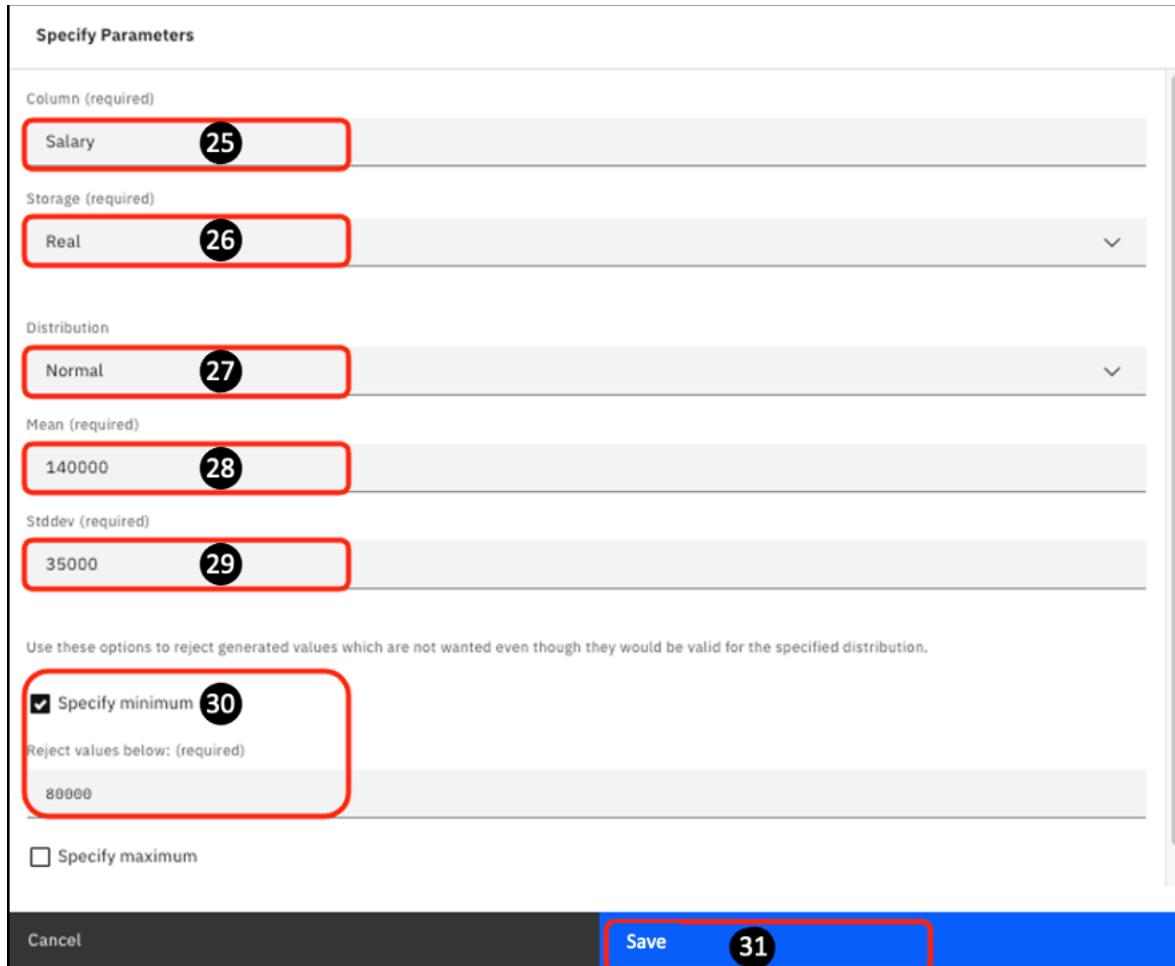
Specify minimum **30**

Reject values below: (required)

80000

Specify maximum

Cancel **Save** **31**



33. You are returned to the **Generate synthetic tabular data flow** page. You now see all 3 columns you defined have been added.
34. Click **Next** to generate the data.

Generate synthetic tabular data flow

Select use case

Generate options **33**

Export data

Review

Generate options

Rows

How many rows would you like to generate

Number of rows
1000 - +
(Min: 1,000; Max: 2,147,283,647)

Columns

Define properties and specify fields, storage types, statistical distributions, and distribution parameters.

Column	Storage	Distrib...	Param...
Last_Name	String	Categorical	[Brown=0...]
Age	Integer	Normal	[mean=50...]
Salary	Real	Normal	[mean=14...]

Add column +

[Close](#) [Back](#) [Next](#) **34**

35. On the next panel, you can specify the output **File name**. Change it to **Name_Age_Salary**

36. Click on the **File Type** drop-down and select **Excel**.

37. Click **Next**.

Generate synthetic tabular data flow

Select use case

Generate options

Export data **35**

Review

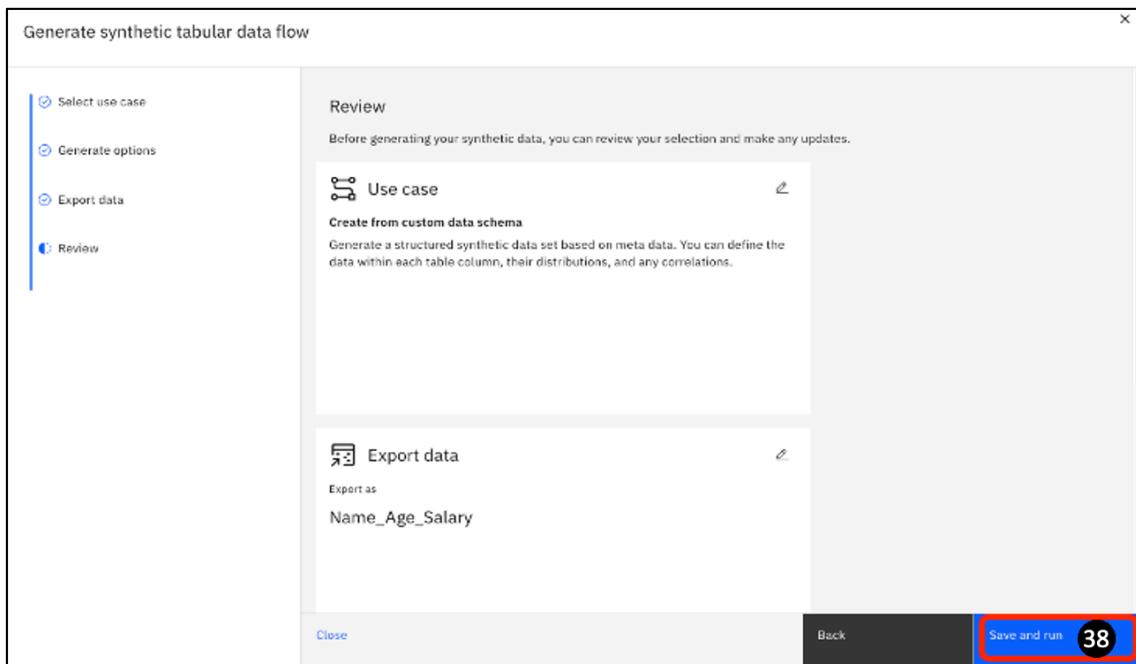
Export data

File name
Name_Age_Salary

File type
Excel **36**

[Close](#) [Back](#) [Next](#) **37**

38. On the next page, review the information. If it is correct, click **Save and run**.



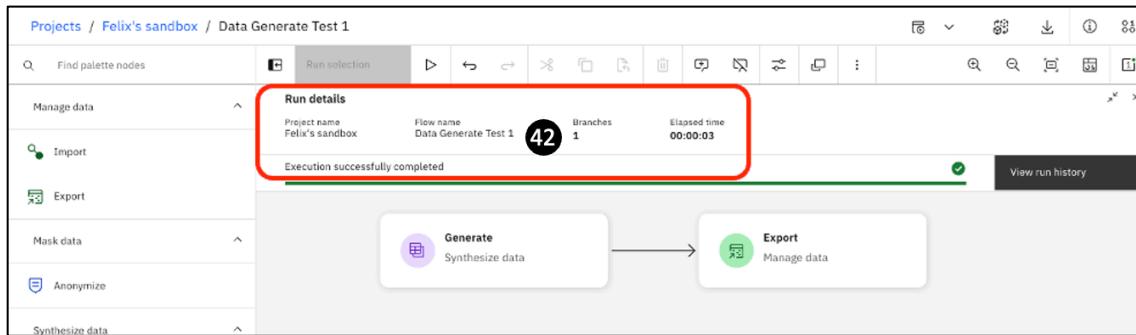
The next page shows a graphical representation of the synthetic data generation tasks. There are 3 main areas:

39. **Generate** – this is where you define how you want to generate your data.
40. **Export** – this is where you define how you want to export your output; in this lab (Step 36) you selected the output to be an Excel file with the name **Name_Age_Salary.xls**.
- When you click on **Generate** and **Export**, you will see that dotted lines will now surround both actions.
41. **Run selection** – you can separately highlight the **Generate** and **Export** icons to run the code individually (you will need to first run **Generate** pipeline before you can run **Export**). In most cases, simply select both of the icons to first **Generate** and then **Export** the generated data in one single pass.

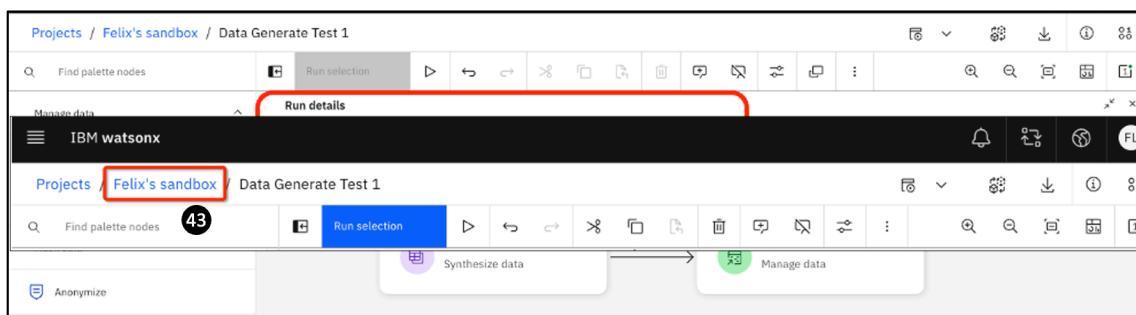


42. The pipeline will take some seconds to run (it can vary depending on the availability of resources – anywhere from several seconds to tens of seconds, so be patient if it takes a little longer).

You will see a quick summary with the **Run details**.

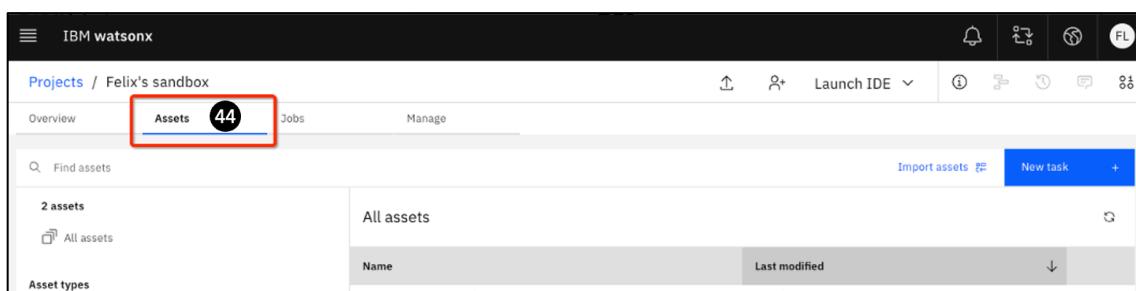


43. Go to the breadcrumb (top left) and click on your project (looks like <your_name>'s sandbox).



Watsonx.ai should open up in your project's **Assets** tab.

44. If not, select the **Assets** tab.



You will see the following (you may have other assets):

45. **Name_Age_Salary.xls** ... the name of the XLS file.

46. **Data Generate Test 1** ... the name of the Synthetic data flow specified in Section 6 Step 5 of this lab.

The screenshot shows the WatsonX AI Platform interface. The top navigation bar includes 'Projects / Felix's sandbox', 'Launch IDE', and various icons. The main area is titled 'Assets' with tabs for 'Overview', 'Assets', 'Jobs', and 'Manage'. A search bar says 'Find assets' and there are buttons for 'Import assets' and 'New task'. On the left, there's a sidebar with 'Asset types' sections for 'Data' and 'Flows', each with a count of 1. The main table lists 'All assets' with columns for 'Name' and 'Last modified'. Two rows are highlighted with red boxes and numbered 45 and 46: 'Name_Age_Salary' (XLS file, last modified 15 minutes ago) and 'Data Generate Test 1' (Synthetic data flow, last modified 15 minutes ago).

Name	Last modified
Name_Age_Salary XLS	15 minutes ago Modified by you
Data Generate Test 1 Synthetic data flow	15 minutes ago Modified by you

6.1.1 Using watsonx.ai to examine and update your synthetic data

One of the key points in generating synthetic data is how closely the data resembles real-life data. It is important to keep in mind that to the AI model, there is no inherent meaning in **Last_Name**, **Age**, and **Salary**. These are simply tokens used. This means that it can sometimes generate data that is “valid” but perhaps makes little sense (or quite unlikely in real life).

In the last section, you generated a **Name_Age_Salary.xls** file with **1000** rows of synthetic data. This data was generated randomly, so your data may not have the same entries as highlighted below; although if you look carefully, the same type of issues are probably present.

Here are some generated data that clearly exhibit various issues.

- There are 2 names with age 0, 8 names with age < 10, and 30 names < 20. Certainly, you would want to remedy this.
- At the other extreme, there are 28 names of age > 75, with 3 of them older than 90. You probably want to also remedy this.
- One more subtle issue is that many older employees are making a lot less than younger employees. This is possible but the distribution simply looks unreal.

In this section, you will look at how to use watsonx.ai to generate better data.

11. Ensure you are in the **Assets** tab of your project and click the **Data Generate Test 1** asset.

12. The Flow page opens. Hover over **Generate** to display 3 vertical dots, then click on them and select **Edit**.

13. Click on the pencil () icon for the **Age** row.

14. The **Specify parameters** page opens. Scroll down to find the **Specify minimum**, and **Specify maximum** checkboxes.

15. Select the **Specify minimum** checkbox.

16. For **Reject values below:** Enter **21**.

17. Select the **Specify maximum** checkbox

18. For Reject values above: Enter 65.

19. Click **Save**.

Specify Parameters

Reject values below: (required)

21

Reject values above: (required)

65

Cancel

Save

4

Scroll down to find
Specify minimum and
Specify maximum

5 6 7 8 9

10. The **Generate** page opens. Click on **Correlation**.

Generate

Synthesized columns

Column	Storage	Status	Locked	Distribution	Parameters	Min,Max
Last_Name	String	Closest	<input type="checkbox"/>	Categorical	[Brown=0.0...	<input type="button"/> <input type="button"/>
Age	Integer	Closest	<input type="checkbox"/>	Normal	[mean=50,s... [max=65,...	<input type="button"/> <input type="button"/>
Salary	Real	Closest	<input type="checkbox"/>	Normal	[mean=140... [max=,mi...	<input type="button"/> <input type="button"/>

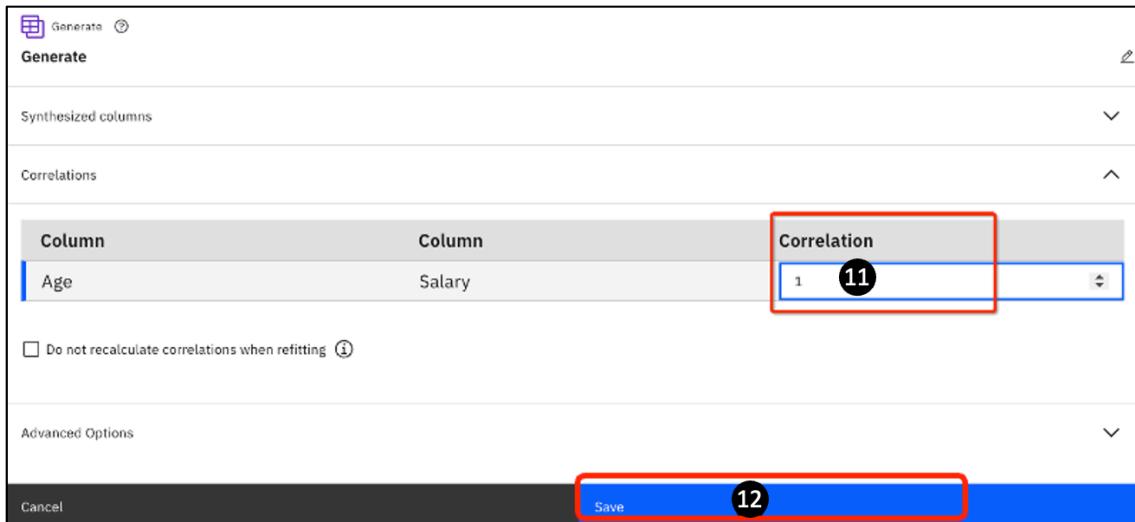
Do not clear Min and Max when refitting (i)

Correlations

10

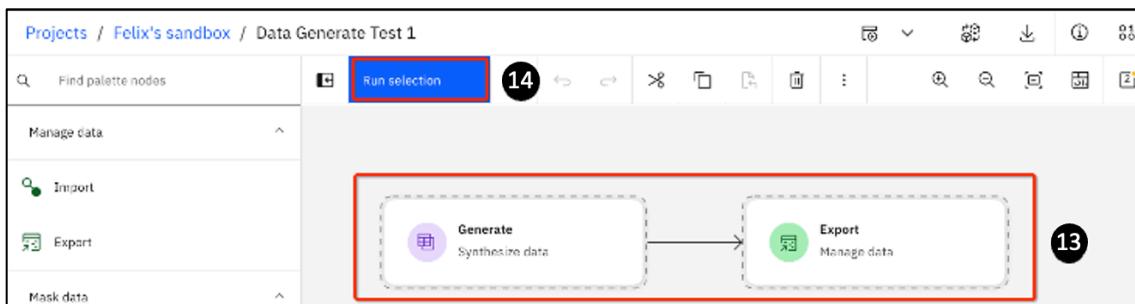
11. Watsonx.ai automatically finds the columns that can correlate: Age and Salary (there is no correlation between a String and a number). Notice that initially the value of **Correlation** is set to 0 – meaning that there is no correlation. Change the value to 1.

12. Click **Save**.



13. The Data Generate Test 1 page opens. Click both Generate and Export.

14. Click Run selection.



15. Click on your project's name on the breadcrumb: <your username>'s sandbox.



16. Your project should open in the Assets tab. If not, select the Assets tab. You should see the newly exported **Name_Age_Salary.xls** file. Click on this file to open it. When you examine the file, the previous issues with employees being too young (< 16) and too old (> 65) are gone.

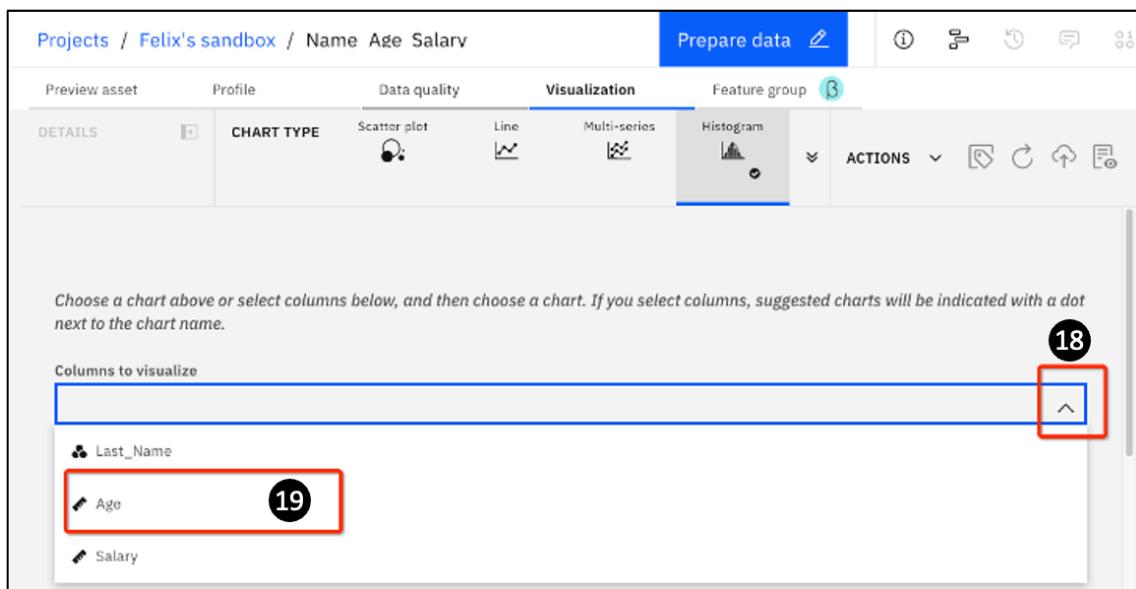
17. Click Visualization on the **Name_Age_Salary** preview page.



Projects / Felix's sandbox / Name Age Salary			Prepare data	⋮
Preview asset	Profile	Data quality	Visualization	Feature group
Preview count: 3 Columns 1000 Rows The preview includes only a limited set of columns and rows.				
Last_Name	Age	Salary		Last refresh: 4 hours ago
Thomas	55	135420.138941		
Leung	57	98837.616656		

18. On the **Visualization** page, click the **Columns to visualize** drop-down.

19. Select **Age**.

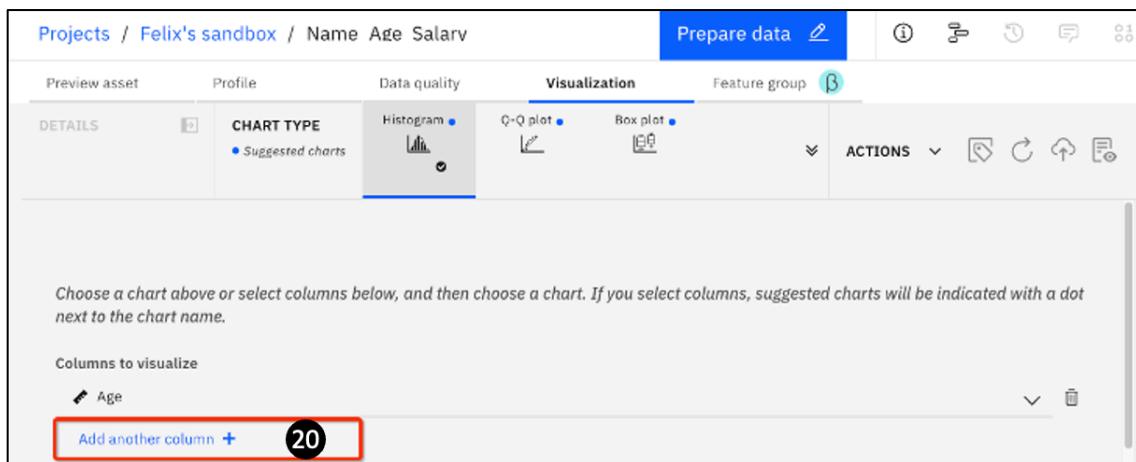


Choose a chart above or select columns below, and then choose a chart. If you select columns, suggested charts will be indicated with a dot next to the chart name.

Columns to visualize

- Last_Name
- Age**
- Salary

20. Click **Add another column +**, then repeat Step 10 to add a **Salary** column:



Choose a chart above or select columns below, and then choose a chart. If you select columns, suggested charts will be indicated with a dot next to the chart name.

Columns to visualize

- Age
- Add another column +

21. You now have both the **Age** and **Salary** columns added.

22. Click **Visualize data**.

Projects / Felix's sandbox / Name Age Salary

Prepare data

DETAILS CHART TYPE Suggested charts Scatter plot

Data quality Visualization Feature group 3

Line Multi-series Parallel Beta

ACTIONS

Choose a chart above or select columns below, and then choose a chart. If you select columns, suggested charts will be indicated with a dot next to the chart name.

Columns to visualize

Age **21**

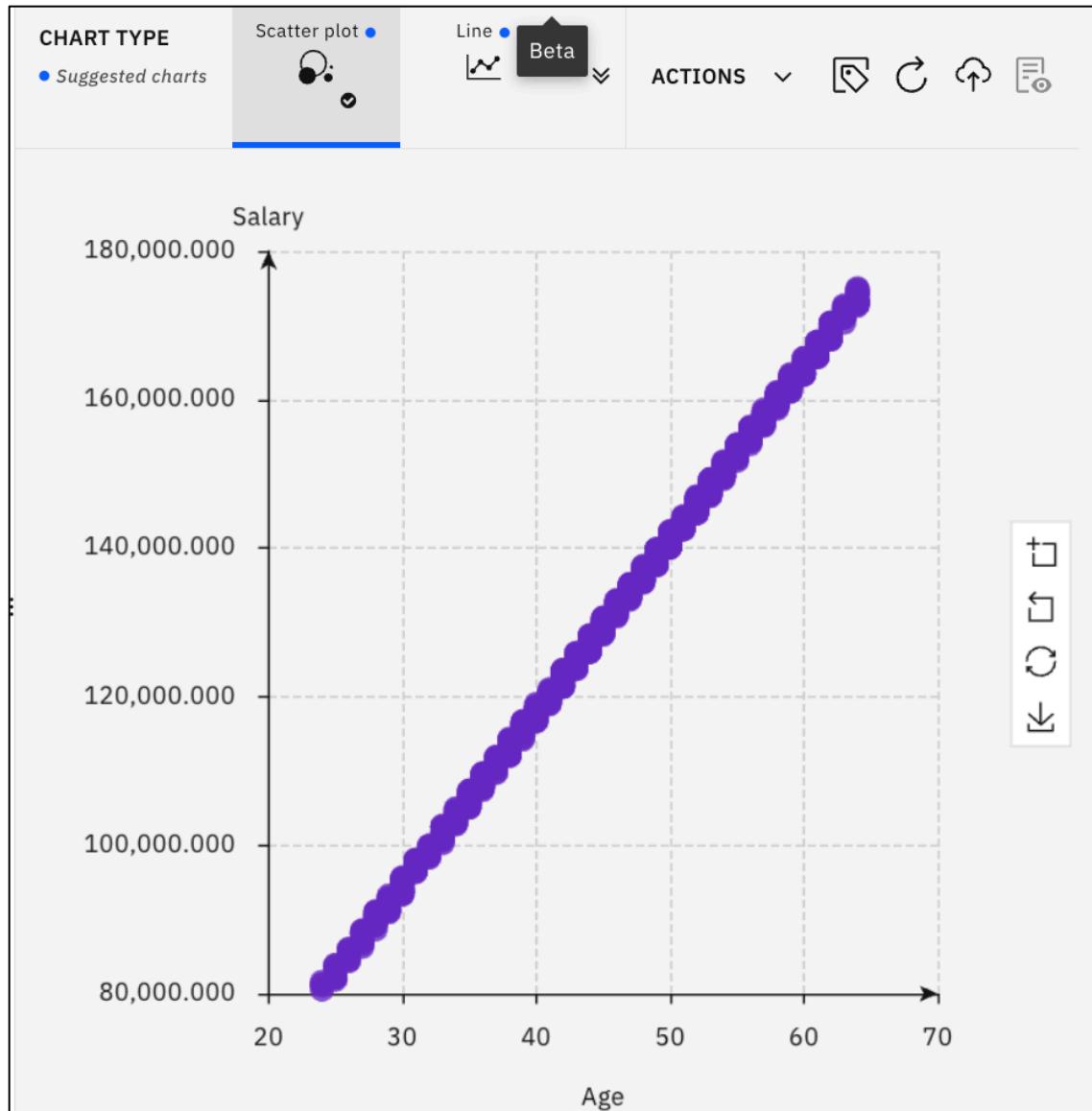
Salary

Add another column

SELECTED COLUMNS **2**

Visualize data **22**

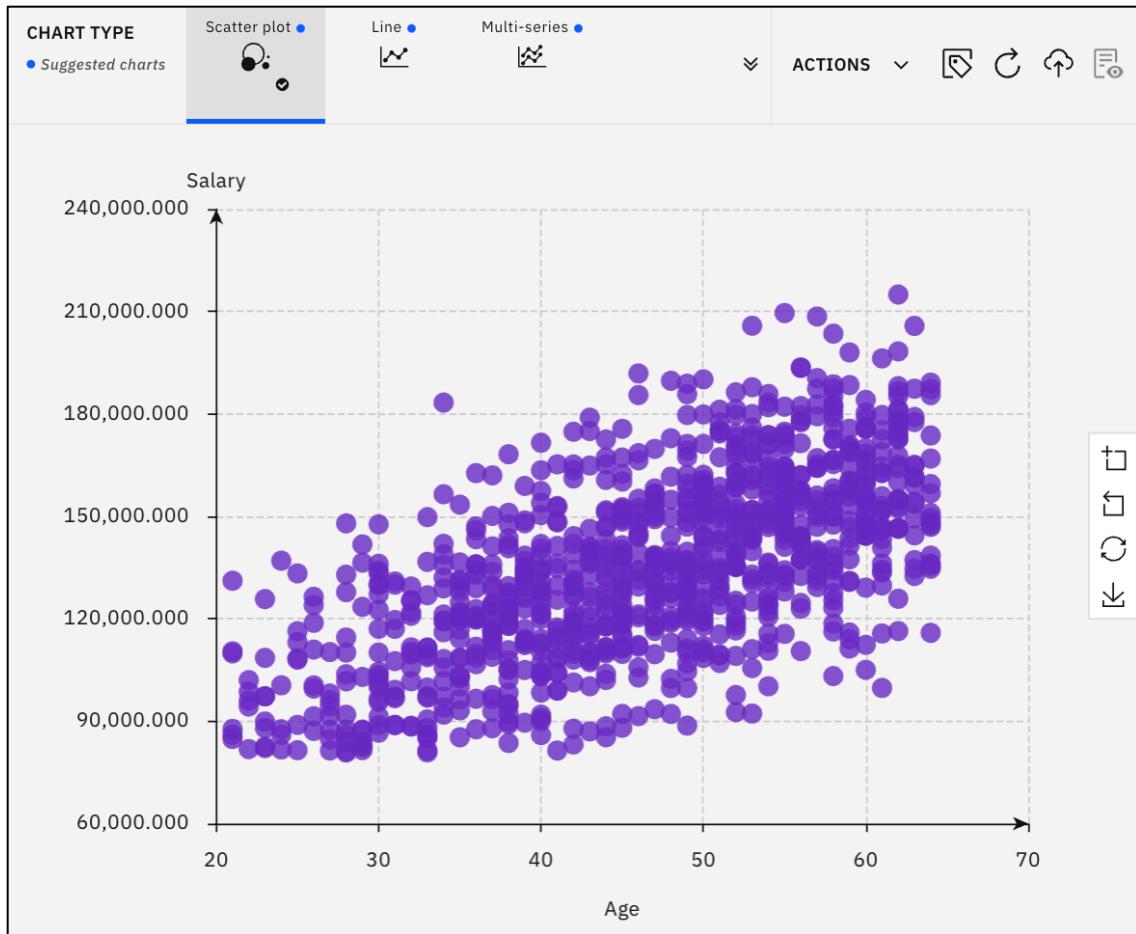
23. You will see something similar to this:



Looking at this though, it does NOT look realistic – because this is a perfect correlation – meaning that the older you get, the higher your salary. While this may be a general trend, experience tells you that it is not realistic. You need something with more variance.

24. You realize that setting the value for **Correlation** to **1** in Step 5 is the problem. To get a more realistic set of data, repeat Steps 5-13. This time setting the value of **Correlation** to **0.8** in Step 5. Run through Steps 6-13 to re-generate and visualize the data again.

Now, you should see a scatter plot looking similar to this:



You can other values such as **0.7** if you want. However, this set of data looks reasonable to be used.

Section summary

In this section, you generated synthetic data to your specifications. IBM watsonx.ai provides a wide range of capabilities – allowing you to control how the data can be generated, including:

- Columns of different types: string, integer, real, time, date, timestamp
- Different data distributions: normal, binary, binomial, and categorical. See Appendix B for more information. This provides the ability to create data that resembles real-life data.
 - You can do “manual” probability assignments to entries like you did in Section 5.1 Step 2. However, keep in mind that as you may be generating hundreds of thousands of rows this can be time-consuming.
- You can define maximum and minimum boundaries for data distribution to reflect real-world data.
- You can define the correlation between different columns to reflect real-world data.

6.2 Create synthetic data by mimicking existing data/schema

Another way to generate a large body of synthetic data is by leveraging existing data. In this section, you will provide a small sample to watsonx.ai and generate a large set of synthetic data that resembles the characteristics of your input data (in distribution, correlation, etc.).

6.2.1 Adding data asset to your sandbox project

1. Open the watsonx.ai console and click on the sandbox project: <your name> sandbox.

The screenshot shows the IBM WatsonX AI console interface. At the top, there's a navigation bar with icons for notifications, search, and user profile. Below it is a welcome message "Welcome back, Felix". A "Customize my journey" button is visible. There are four main cards: "Experiment with foundation models and build prompts with Prompt Lab", "Build machine learning models automatically with AutoAI", "Work with data and models in Python or R notebooks with Jupyter notebook editor", and "Prepare and visualize data with Data Refinery". Below these are sections for "Jump back in" (Recently visited pages), "Recent work" (Projects and Deployment spaces), and "Data in this project" (Drop data files here or browse for files to upload).

2. Download this data asset from this [link](#) as a seed file. Store on a local directory. Add the file to your project by clicking on the **Drop data files here or browse for files** link to upload the **titanic.csv** file from a local directory.

The screenshot shows the "Assets" tab in the "Felix's sandbox" project. It has tabs for "Overview", "Assets" (which is selected), "Jobs", and "Manage". There are buttons for "Import assets" and "New asset". The main area shows "0 asset" and a "Find assets" search bar. To the right, there's a sidebar titled "Data in this project" with a red box around the "Drop data files here or browse for files to upload" link, which is circled with number 2.

3. The sandbox project page opens. You will see that this file shows up in the **Assets** tab.

The screenshot shows the IBM Watsonx interface with the 'Assets' tab selected. On the left, there's a sidebar with '1 assets' and 'All assets'. The main area displays a table of assets under 'All assets' with columns for 'Name' and 'Last modified'. The asset 'titanic.csv' is highlighted with a red box and circled with number 3.

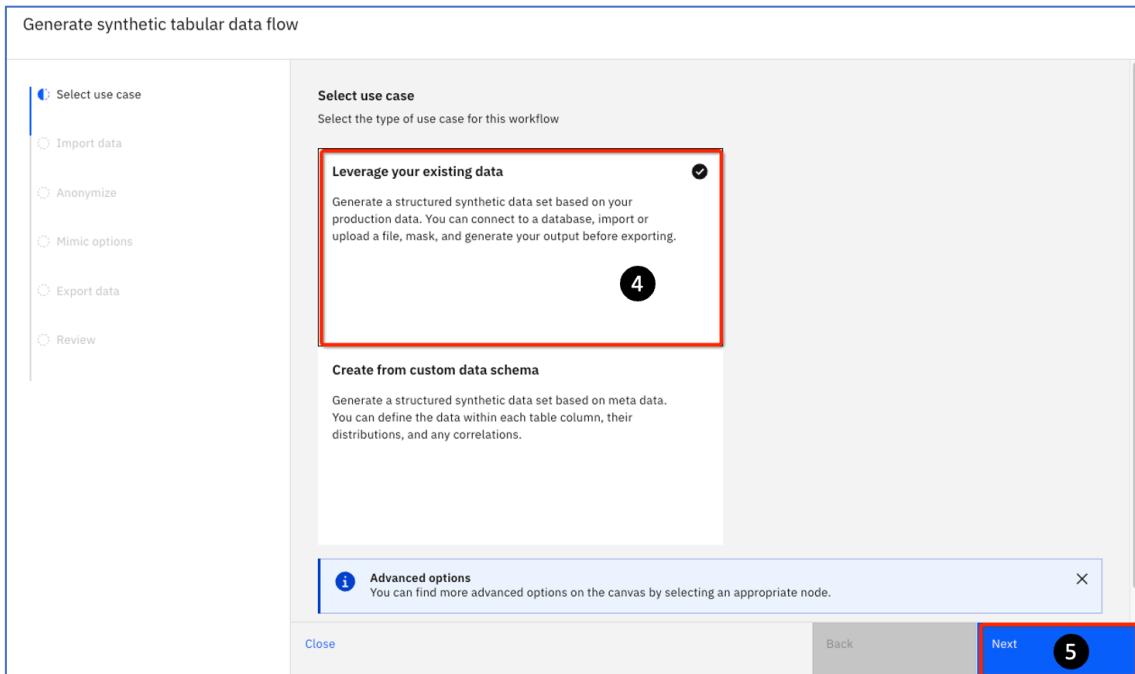
6.2.2 Generating data with a seed file

Repeat Steps 1 – 5 from Section 5.1. For Step 5, use the following values:

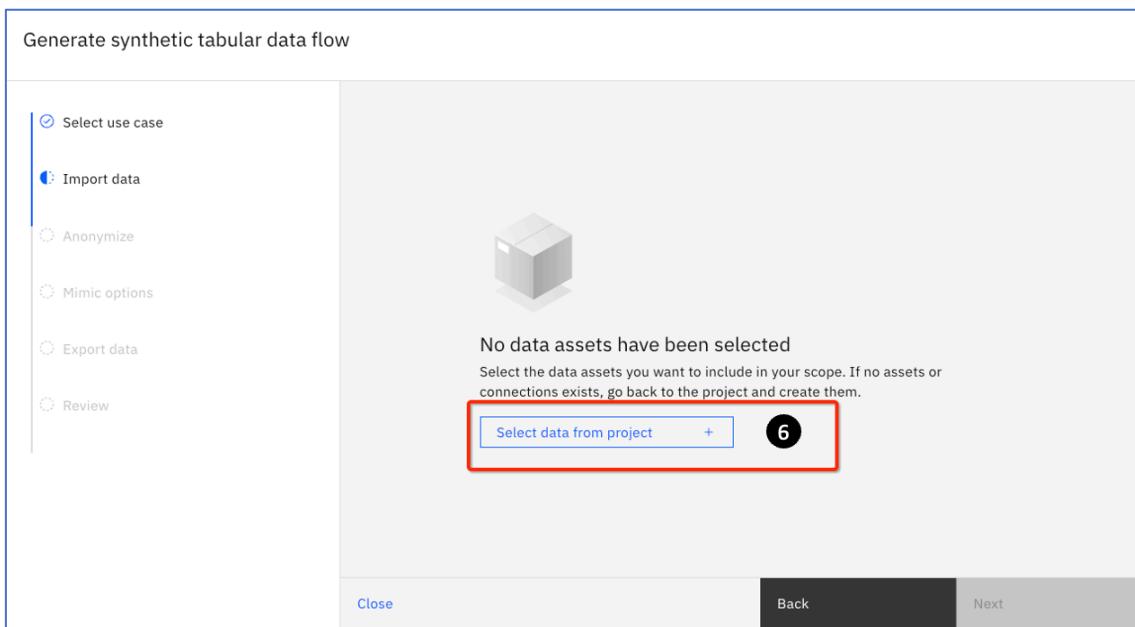
1. For Name: Enter **Sales data generation with seed**
2. For Description: Enter **Generate a large synthetic sales data set using a seed file**
3. Click **Generate**.

The dialog box is titled 'Generate synthetic tabular data' and contains fields for 'Name' and 'Description'. The 'Name' field is set to 'Sales data generation with seed' (circled with 1) and the 'Description' field is set to 'Generate a large synthetic sales data set using a seed file' (circled with 2). The 'Create' button at the bottom right is highlighted with a red box and circled with 3.

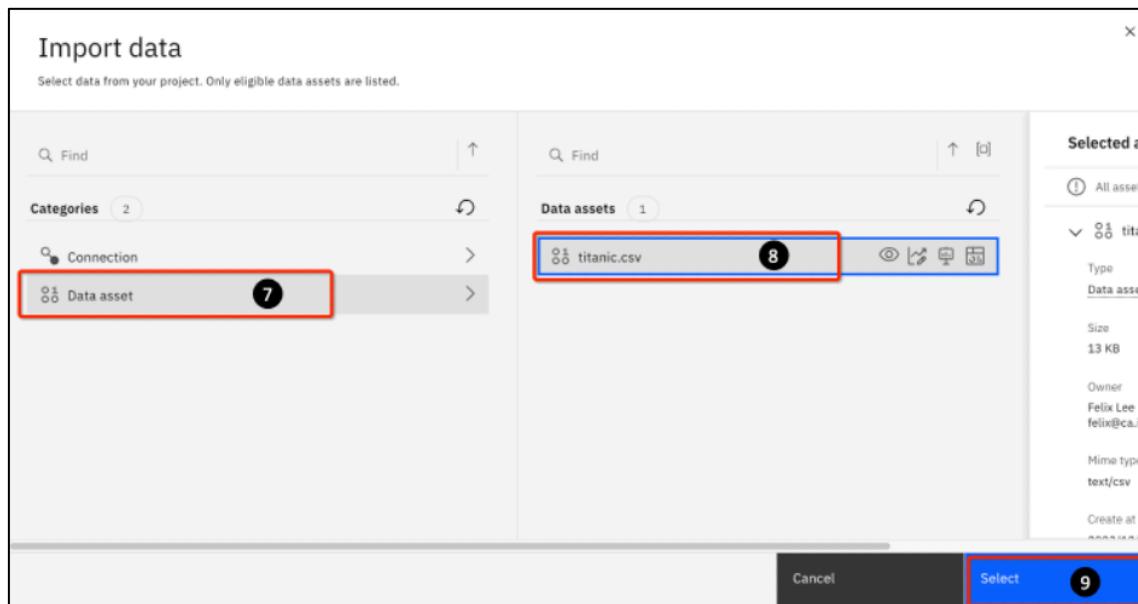
4. Repeat Steps 6 – 9 from Section 5.1. Only this time for Step 9, select the **Leverage your existing data** use case on the **Select use case** tab.
5. Click **Next**.



6. The focus is changed to **Import data** task. Click on **Select data from project**.



7. Click on **Data asset**. This will slide open a pane on the right.
8. Select the **titanic.csv** file.
9. Click **Select**.



10. You are returned to the **Generate Synthetic tabular data flow** page. Note that the **titanic.csv** file has now been imported. Click **Next**.

Asset name	Last modified	Created on
titanic.csv	48 minutes ago	48 minutes ago

Size limit notice
This environment can import up to ~2.5GB of data. If you receive a related error message or your data fails to import, please reduce the amount of data and try again.

11. Focus now changed to the **Anonymize** task, options are provided to determine if any columns need to be anonymized. For this lab, anonymization is not required. Click **Next**.

Anonymize columns

You can disguise column values when working with data.

Column name	Anonymize
PassengerId	No ▾
Survived	No ▾
Pclass	No ▾
Sex	No ▾
Age	No ▾
SibSp	No ▾
Parch	No ▾
Fare	No ▾
Cabin	No ▾
Embarked	No ▾

[Close](#) [Back](#) [Next](#) **11**

12. This tab lets you some **mimic** options. In **mimic**, the synthetic data generated takes on the characteristics of the input data (in terms of data type, distribution, etc.) By default, watsonx.ai will generate 100,000 rows. For this exercise, change **Number of rows** to the minimum number allowed: 1,000.
13. For the **Goodness of fit** criteria, use the default **Kolmogorov-Smirnov**.

Both **Kolmogorov-Smirnov** and **Anderson-Darling** are statistical tests that measure how close are two sets of data (the seed data and the one being generated).

Kolmogorov-Smirnov is a better test for the more sensitive data around the center of the data distribution, whereas Anderson-Darling is more sensitive to the tails.

For this exercise, simply take the default.

14. Click **Next**.

Generate synthetic tabular data flow

Mimic options

Number of rows
1000 **12** - +
(Min: 1,000; Max: 2,147,283,647)

Goodness of fit criteria (continuous fields only) [\(i\)](#)

Kolmogorov-Smirnov **13**
 Anderson-Darling

[Close](#) [Back](#) [Next](#) **14**

15. For File Name: Enter **mimic-output**

16. For File type, keep the default value of CSV

Note: There are other formats like - delimited, Excel, JSON, parquet, SAV, XML).

17. Click **Next**.

Generate synthetic tabular data flow

Export data

File name
mimic-output.csv **15**

File type
CSV **16** ▾

[Close](#) [Back](#) [Next](#) **17**

18. Focus is now on the **Review** task. Click the **Edit** icon () repeatedly to step through Steps 5 through 17. Review the information.

Use case: Leverage your existing data

Import data: titanic.csv

Number of columns:12

Anonymize: 0

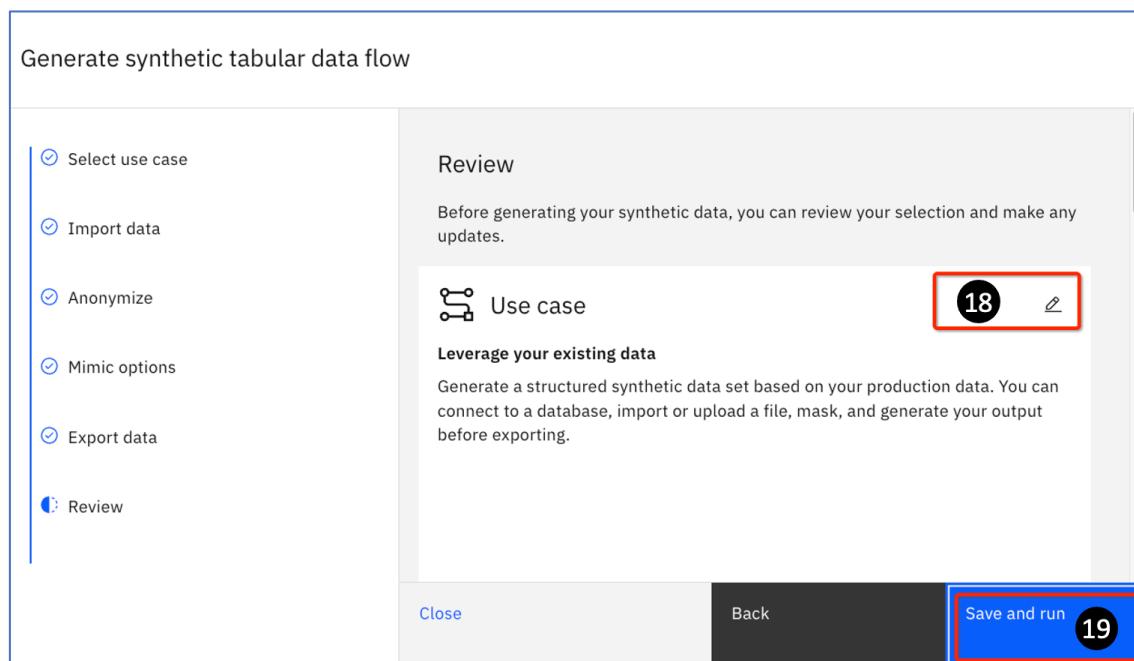
Mimic options:

Number of rows: 1000

Goodness of fit: Kolmogorov-Smirnov

Export data: Export as: mimic-output.csv

19. After verification click on **Save and run**.

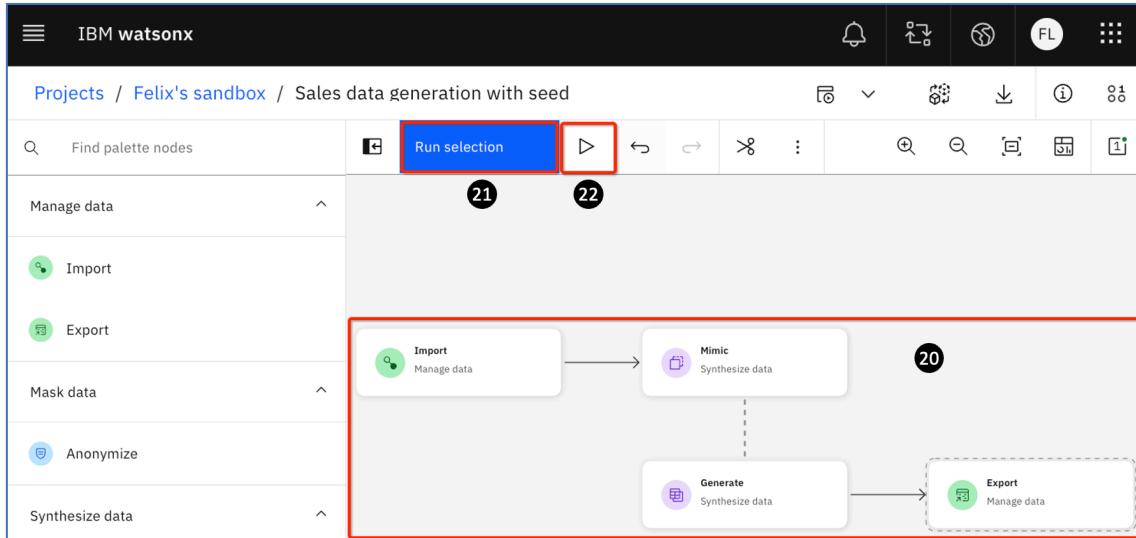


20. The next page shows a graphical representation of the tasks. There are 4 main areas:

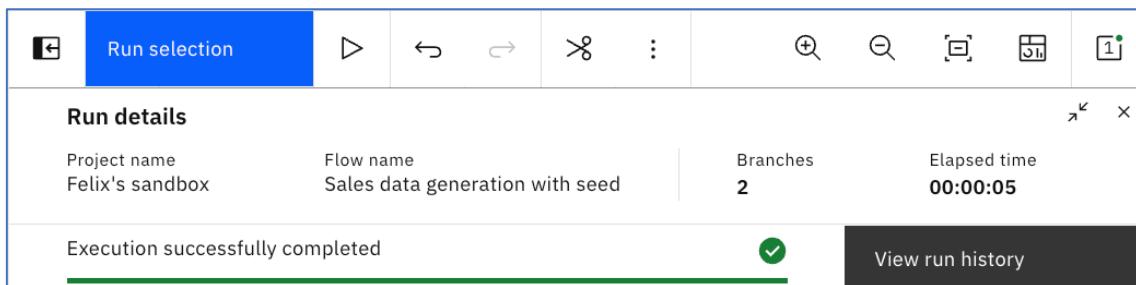
- **Import** – this is where you define how you want to generate your data.
- **Mimic** – this is where the data is mimicked.
- **Generate** – this generates data.
- **Export** – this is where you define how you want to export your output – in this example, this would be in a CSV file named **output.csv**.

21. **Run selection** – you can choose to run Generate on its own (to do so, click the **Generate** tile before clicking **Run** selection) before running the **Export** pipeline; or you can choose to run both steps in one pass.

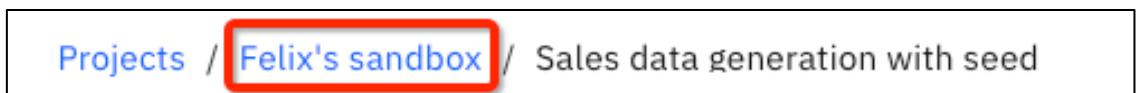
22. Click the **Play** icon to run all nodes in the pipeline.



23. The pipeline will take some seconds to run (it can vary depending on the availability of resources – anywhere from several seconds to tens of seconds, so be patient if it takes a little longer). You will see a quick summary as follows:



24. Go to the breadcrumb and click your project (likely <your_name>'s sandbox).



25. You will see the following (you may have other assets):

- CSV: is the output file **mimic-output.csv**
- Synthetic data flow is the name you used: Sales data generation with seed
- The original file – **titanic.csv**

Click on **mimic-output.csv** to look at the newly generated dataset. This contains 1000 rows of synthetically generated data.

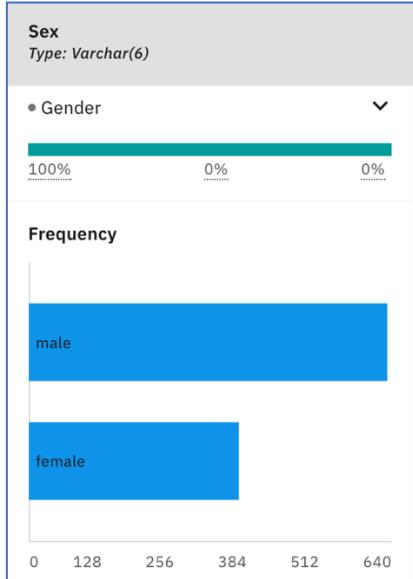
The screenshot shows the IBM Watsonx interface with the 'Assets' tab selected. On the left, there's a sidebar for 'Asset types' with 'Data' and 'Flows' sections. The main area displays a table titled 'All assets' with columns for 'Name', 'Last modified', and a sorting arrow. The first item, 'mimic-output.csv', is highlighted with a red box and has a black circle with the number '25' over it, indicating the number of rows. Other items listed include 'Sales data generation with seed' (modified 15 minutes ago) and 'titanic.csv' (modified 5 hours ago).

26. The output should look something like this:

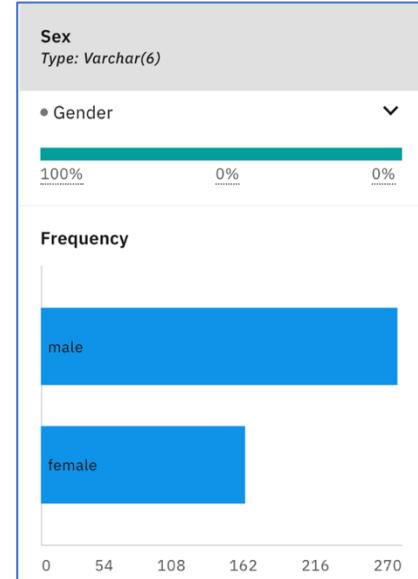
PassengerId	Survived	Pclass	Age	SibSp	Parch	Fare	Sex	Cabin	Embarked
1116	0	1	33.270904	1	1	49.675491	male	A21	S
1076	0	1	42.687557	0	1	60.103383	male		S
1114	0	2	5.77919	1	1	21.437806	female		S
1197	0	0	40.851028	0	0	88.615865	male		S
1265	0	2	29.030022	1	1	29.129059	female		Q
969	0	1	23.4798	1	0	53.02443	female	C54	S
1130	0	2	31.980727	0	0	15.564615	female		S
1195	0	1	28.946602	1	-1	22.435221	female		S

27. Now let's compare the distribution of the original data set and the generated dataset. As you can see, the distributions for gender/sex are similar for both the generated dataset and the original dataset.

Generated dataset

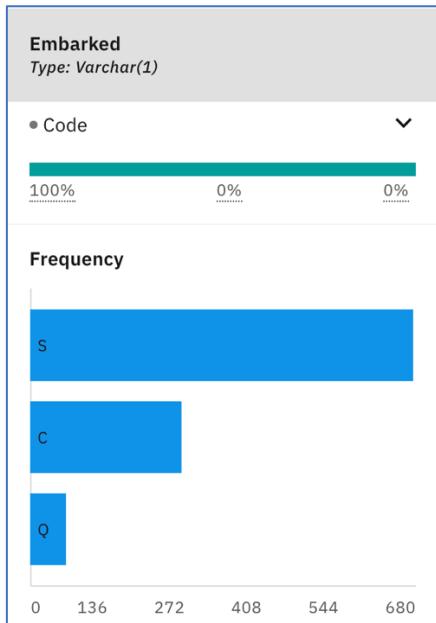


Original dataset

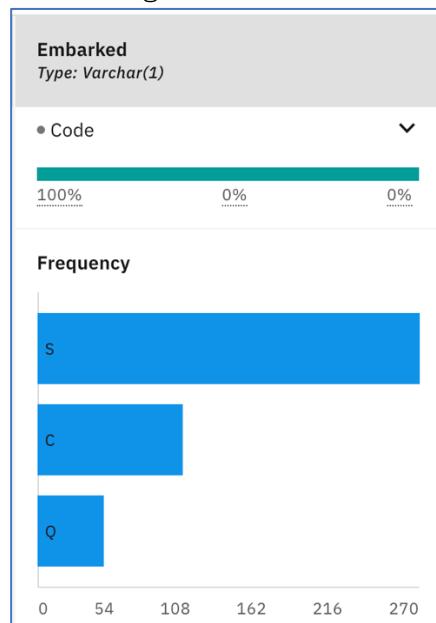


Similarly, the distribution of **Embarked** looks similar.

Generated dataset



Original dataset



Section summary

In this section, you generated synthetic data using an existing dataset. IBM watsonx.ai provides a wide range of capabilities – allowing you to control how the data can be mimicked, including:

- The number of data points that you want to generate.
- Different methods to generate data - Kolmogorov-Smirnov and Anderson-Darling
- You can also choose to anonymize some columns.
- You can profile the datasets and build correlations between different columns to reflect real-world data.

The advantage of generating data using a seed file is that there is no need to manually define the data schema. Its distribution will also closely resemble that of the input data. If the input data is a good representation of the overall dataset, the generated data will also resemble the client's business reality.

Appendix A. Revision History

Date	Changes
-	Original version.