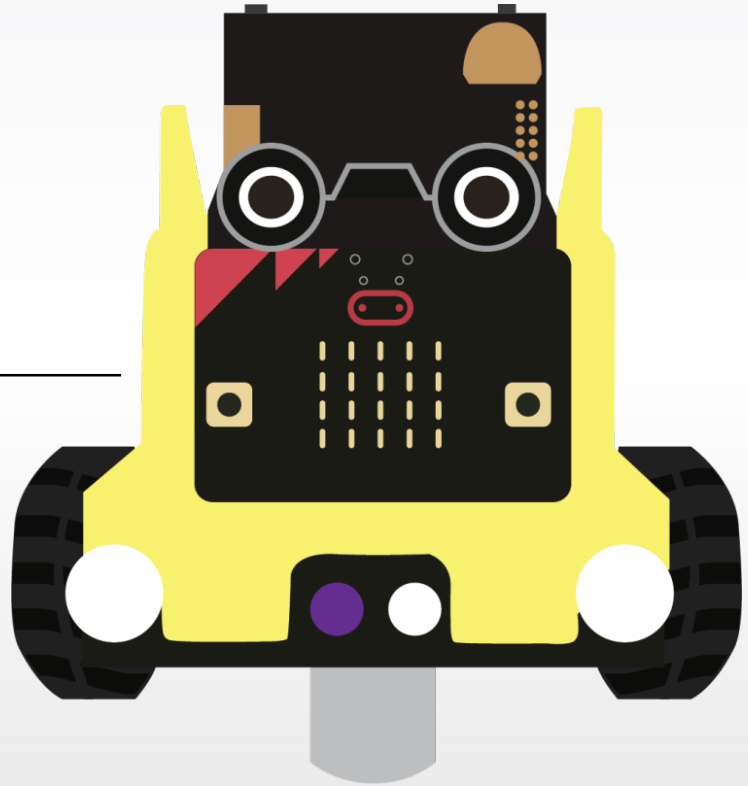
AI for the Future
CUHK Jockey Club AI for the Future Project
中大賽馬會智為未來計劃

AI with Python



Teacher Workshop

Dr. Xing Xing (Symphony)
Mudit Chaudhary

Co-organized by:
The Chinese University of Hong Kong
Faculty of Engineering
Faculty of Education

Funded by:
The Hong Kong Jockey Club Charities Trust

Content

CUHK Jockey Club AI for the Future Project
中大賽馬會智為未來計劃

Getting Started

1. What is Python?

Python Basics

1. Hello World
2. Programming concepts (Variables, If-else statements, Loops, Lists, Dictionaries, Functions, and Libraries)

AI Project

1. Create a virtual assistant (tell joke, get weather, get movie rating, get movie director, get movie cast)

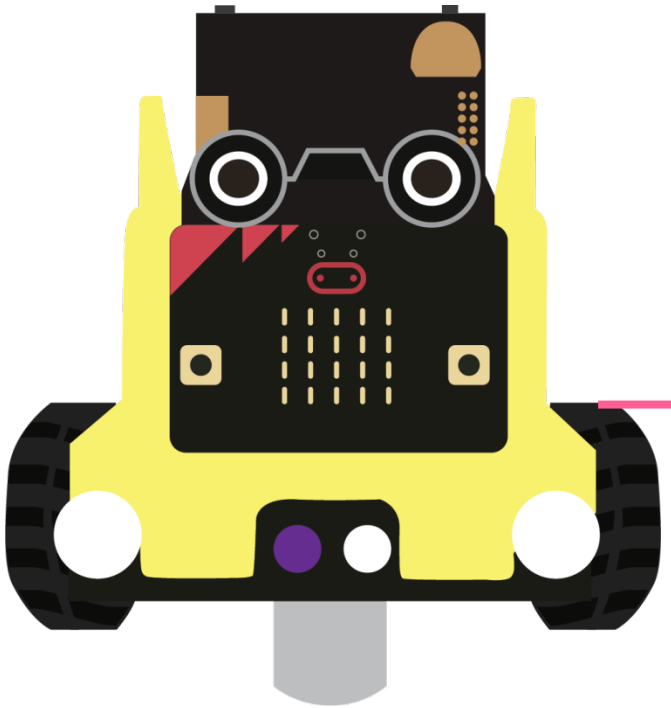
14:00 ~ 15:30
Lecture & Jupyter notebook 1

15:30 ~ 15:45
Breakout room/Take break

15:45 ~ 17:15
Lecture & Jupyter notebook 2
Survey

17:15 ~ 17:30
Breakout room/Early leave

CUHK Jockey Club AI for the Future Project
中大賽馬會智為未來計劃

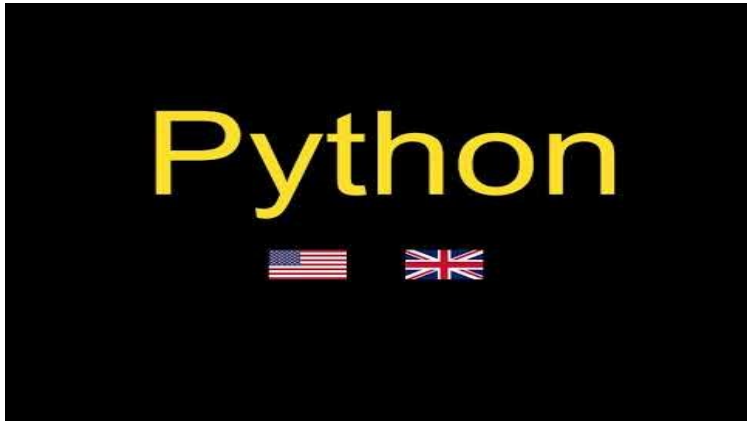


Getting Started

What is Python?

- python™ is a programming language created by **Guido van Rossum** in 1991.
- Python is one of the most popular programming languages for developing **Artificial Intelligence applications.**

Snake genus



Why Python?

- Beginner-friendly
- Clear syntax
- Short code
- Wide range of applications
  - Artificial intelligence/Data science/Machine learning, Web applications, Desktop software, Mobile apps, Big data, Games, …
- Lots of support (e.g., libraries, communities)
- In high demand

Top Programming Languages 2021

Rank	Language	Score
1	Python	100
2	Java	95.4
3	C	94.7
4	C++	92.4
5	JavaScript	88.1
6	C#	82.4
7	R	81.7
8	Go	77.7
9	HTML	75.4
10	Swift	70.4

Source: IEEE

Get Python

CUHK Jockey Club AI for the Future Project
中大賽馬會智為未來計劃

Get Python on your computer:

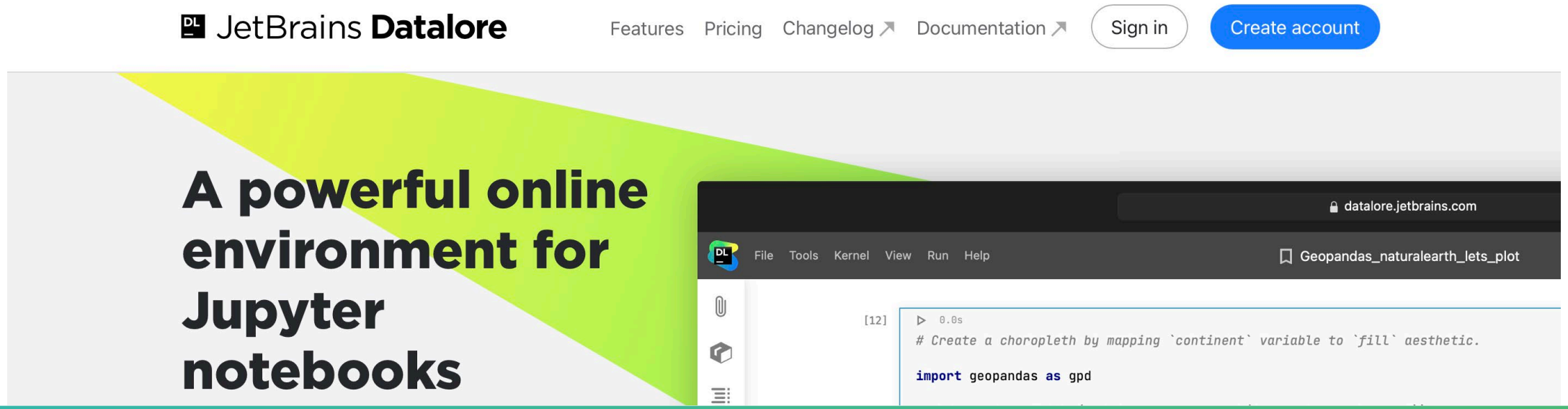
Cloud-based platform:

Introduction to JetBrains Datalore

CUHK Jockey Club AI for the Future Project
中大賽馬會智為未來計劃

In this Workshop, we will use JetBrains **Datalore** to learn, develop, and run Python applications online.

Access and sign in to JetBrains **Datalore** Platform using the following link: datalore.jetbrains.com

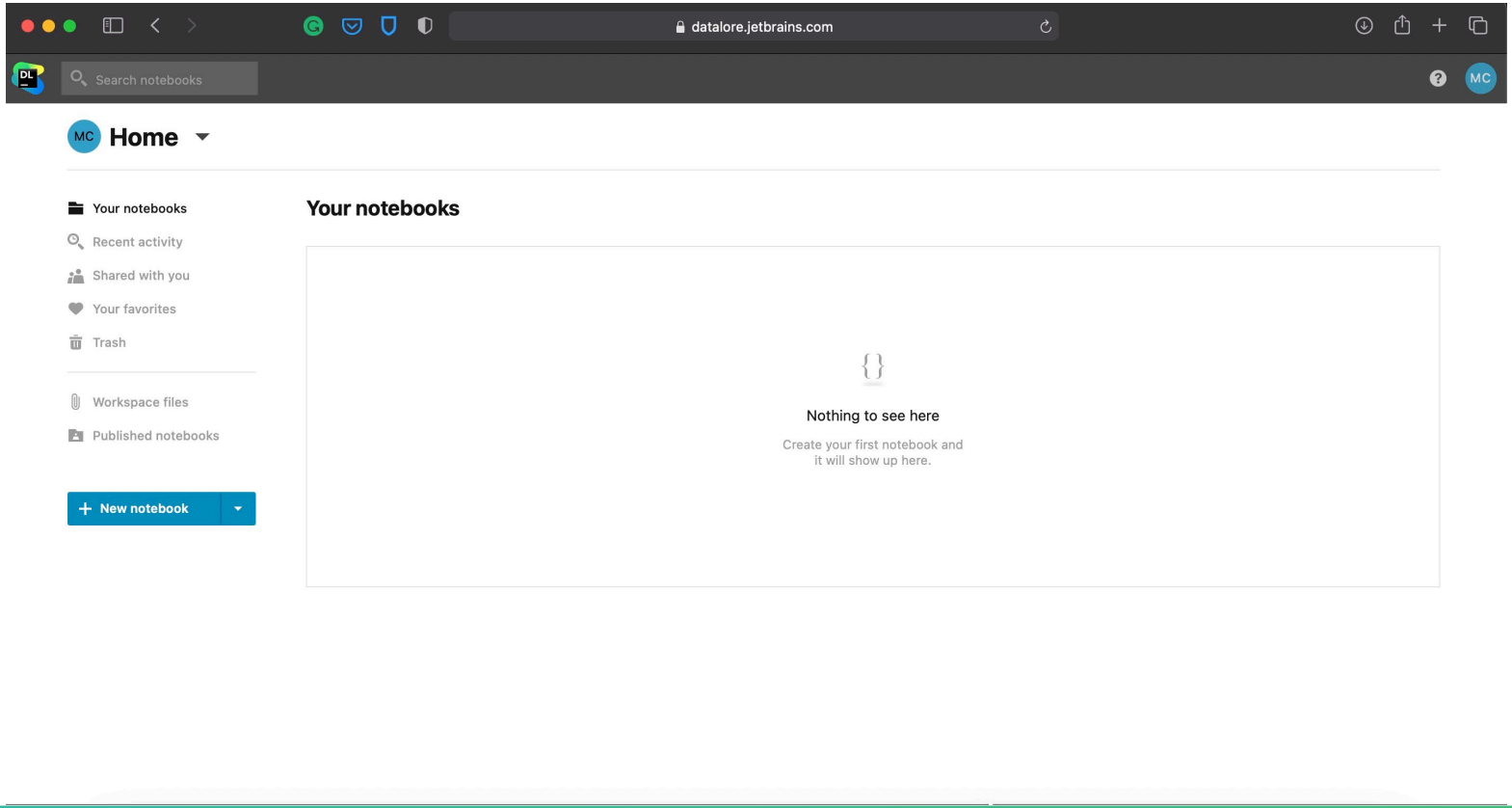


Introduction to JetBrains Datalore

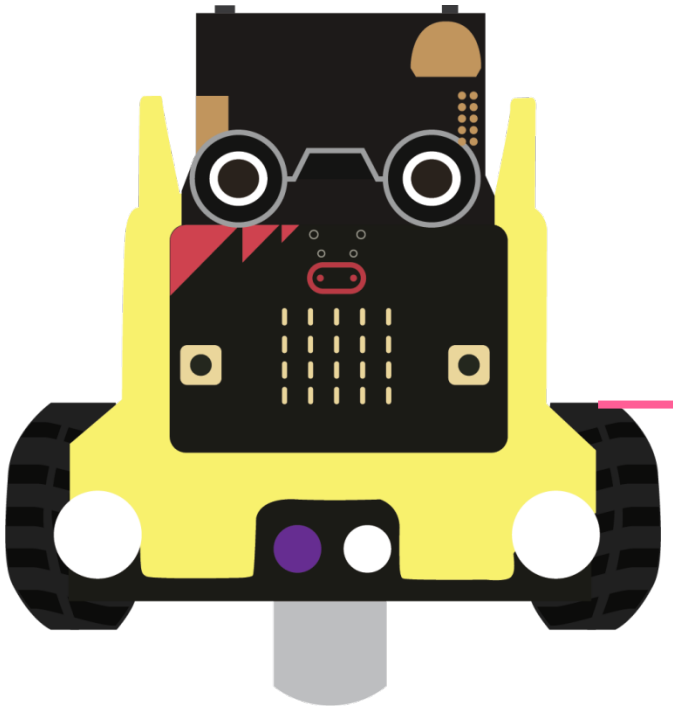
CUHK Jockey Club AI for the Future Project
中大賽馬會智為未來計劃

1. Open our tutorial notebook by pressing the arrow beside 'New Notebook' and select 'Upload notebook'

2. Go to your browser and go to cutt.ly/jHqHzun in the URL bar. Copy the full URL into DataLore and press 'Upload' button



CUHK Jockey Club AI for the Future Project
中大賽馬會智為未來計劃



Python Basics

Content

CUHK Jockey Club AI for the Future Project
中大賽馬會智為未來計劃

Getting Started

1. What is Python?

Python Basics

1. Hello World
2. Programming concepts (Variables, If-else statements, Loops, Lists, Dictionaries, Functions, and Libraries)

AI Project

1. Create a virtual assistant (tell joke, get weather, get movie rating, get movie director, get movie cast)

14:00 ~ 15:30
Lecture & Jupyter notebook 1

15:30 ~ 15:45
Breakout room/Take break

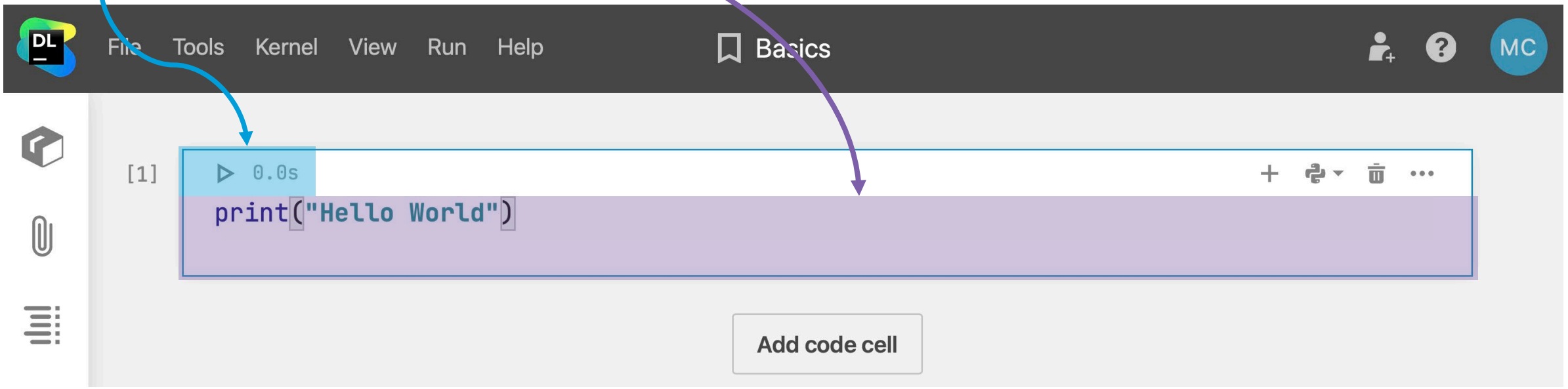
15:45 ~ 17:15
Lecture & Jupyter notebook 2
Survey

17:15 ~ 17:30
Breakout room/Early leave

Hello World

Now, we write our first code in Python. We will print “Hello World” using Python.

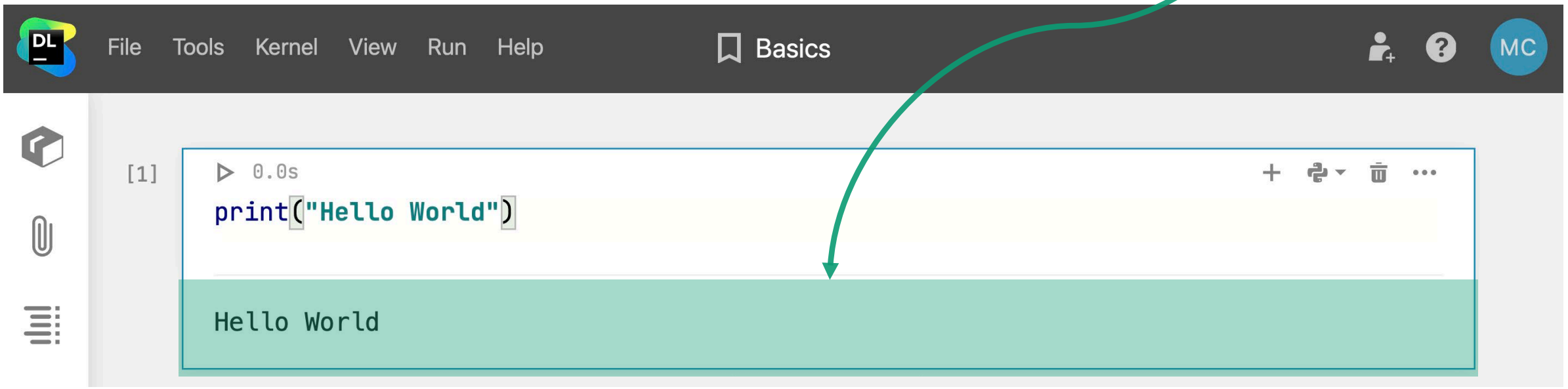
Inside the **coding cell**, type `print(“Hello World”)` and press the **run button.**



Hello World

Once you press the **run button**, you can see the **output** of the code below the **coding cell**.

This is how we will interact with the Jupyter Notebook. We write the code, press **run button**, and then see its output.



Variables

CUHK Jockey Club AI for the Future Project
中大賽馬會智為未來計劃

Variables act like containers which can store values such as numbers, strings, characters, etc.

You can treat **variables** in programming as the variables we use in mathematics.

Let's say we want to assign a variable called **name** with your name. So, how do we do that?

Answer:

```
name = "<Your name here>"
```

Variables

Let's try these exercises on Datalore to get familiar with the concept of **variables**.

1. Create a variable called name to store your name. Use print statement to print it on screen.
2. Create 3 variables **x**, **y**, and **z**. Store their values as follows: **x** = 3; **y** = 7; **z** = **x** + **y**. Print the value of **z** and verify if it is correct.

Data Types

CUHK Jockey Club AI for the Future Project
中大賽馬會智為未來計劃

Data Types: Variables can store data of different types, and different types can do different things.

There are various **Data Types** but the most common data types are:

1. **Integer** – Whole numbers (E.g., 1, 400, -999, etc.)
2. **Float** – Decimals (E.g., 3.67, -4.7, 9.42, etc.)
3. **String** – Collection of characters (E.g., "Hello", "Good morning", "Amazing", etc.)
4. **Boolean** – True, False

Operators

CUHK Jockey Club AI for the Future Project
中大賽馬會智為未來計劃

Operator	Name	Example
+	Addition	x + y
-	Subtraction	x - y
*	Multiplication	x * y
/	Division	x / y
%	Modulus	x % y
**	Exponentiation	x ** y
//	Floor division	x // y

Relational Operators

CUHK Jockey Club AI for the Future Project
中大賽馬會智為未來計劃

Relational Operators are used for comparing values.

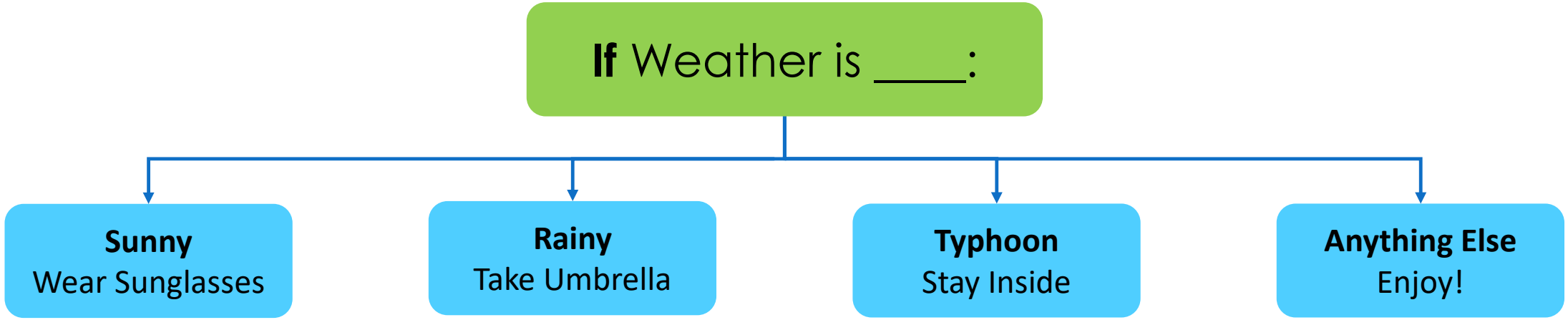
Commonly used relational operators are:

1. **==** (Equal to)
2. **<** (Less than)
3. **>** (Greater than)
4. **>=** (Greater than or equal to)
5. **<=** (Less than or equal to)
6. **!=** (Not equal to)

Let's try a small exercise to get familiar with relational operators in Python

Conditional Statements

- **Conditional Statements** are statements that can control the flow of the program.



Conditional Statements

CUHK Jockey Club AI for the Future Project
中大賽馬會智為未來計劃

One of the conditional statements is called if-else statement. It consists of the following parts:

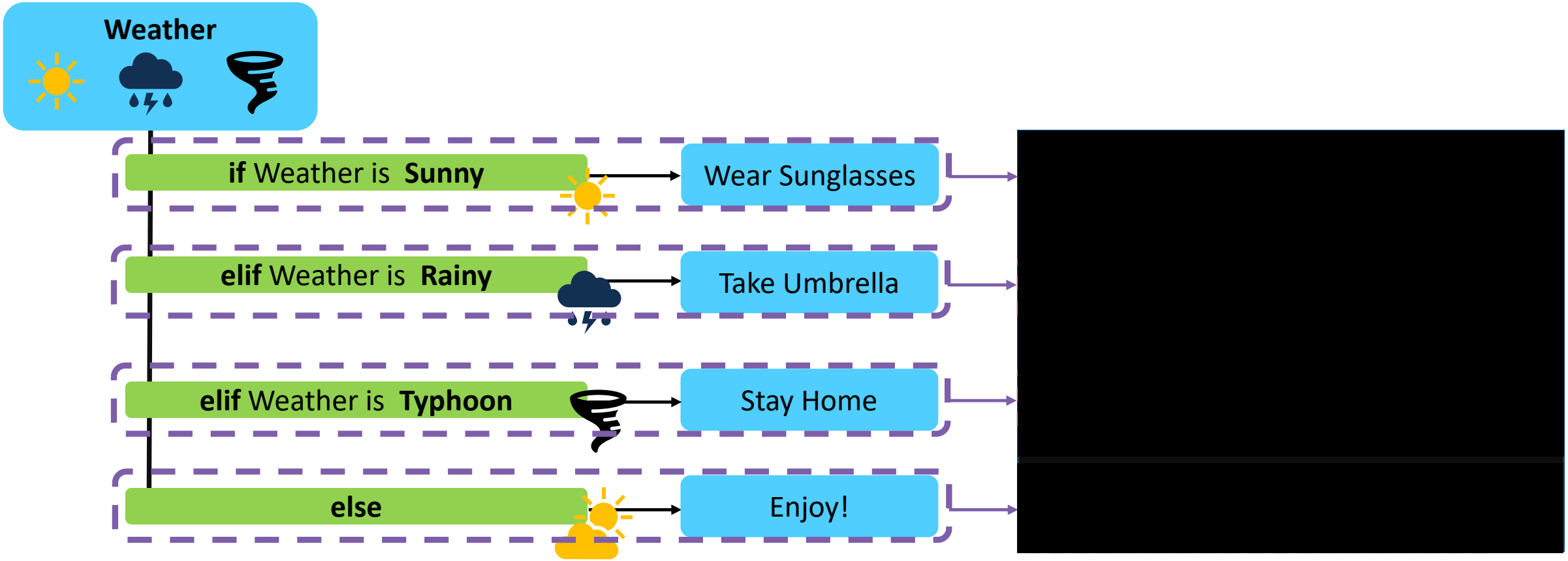
- **if**: All if-else conditional statements have to start with an "if"
- **elif**: Else if, there can be many elif
- **else**: If no other condition is fulfilled, else condition takes over

```
if (Condition 1):
    Statements_1
    ...
elif (Condition 2):
    Statements_2
    ...
elif (Condition 3):
    Statements_3
    ...
...
else:
    Statements_n
    ...
```

Note: Python use indentation to indicate a new code block.

Conditional Statements

Let's code the real-life example we saw earlier in Python and play with it to get familiar with if-else statements.



Conditional Statements

老婆給當程式師的老公打電話：“下班順路買一斤車厘子帶回來，如果看到賣西瓜的，就買一個。”

當晚，程式師老公拿着一個車厘子進了家門……

老婆怒道：“你怎麼就買了一個車厘子？！”

老公答曰：“因為看到了賣西瓜的。”

List

List is used to store multiple items in a single variable.

For example, we have a shopping list with multiple items – Grapes, Apple, Butter, Cheese.

We can denote this shopping list as a list in Python like this –

```
Shopping_list = [“Grapes”, “Apple”, “Butter”, “Cheese”]
```

List


```
Shopping_list = ["Grapes", "Apple", "Butter", "Cheese"]
```

To access an item of a list, we use **index**. In Python, index starts from 0, which is the first element of the list.

To access the first element of the Shopping_list, we have to input:

`Shopping_list[0]`, where 0 is the **index.**

Can you guess the index of "Cheese"?

List

Let's try these exercises on Datalore to get familiar with the concept of **List**.

1. Create a python list called shopping_list and print the last element of the shopping list.
2. Change the last element of the list to "Laptop" and print the whole list.

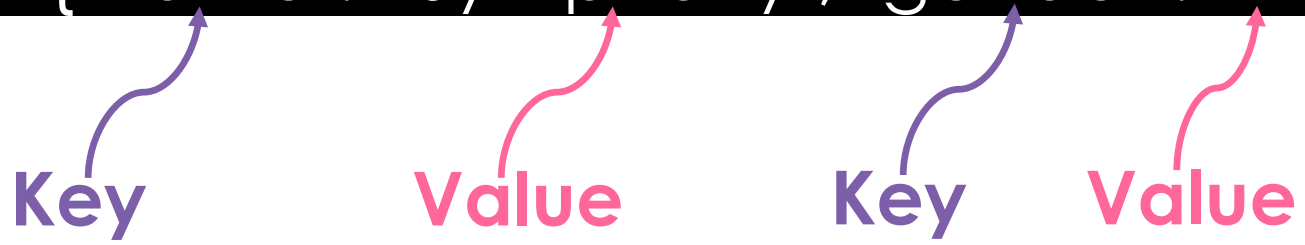
Dictionary

CUHK Jockey Club AI for the Future Project
中大賽馬會智為未來計劃

Dictionary also can store multiple items in a single variable, but items are stored as **key**: **value** pairs.

Dictionary in Python is defined as follows:

```
example_dictionary = {"name": "Symphony", "gender": "F"}
```



To access value of any key from a dictionary, we follow this syntax:

`example_dictionary[key]` E.g., `example_dictionary["name"]`

Dictionary

CUHK Jockey Club AI for the Future Project
中大賽馬會智為未來計劃

Let's try these exercises on Datalore to get familiar with the concept of **Dictionary**:

1. Create a dictionary with your information (keys) – **name**, **age**, **gender**, **major**
2. Print this dictionary
3. Access the value for the key **name**

Loop

Loop is for repeating the same code block multiple times. Loop can make code shorter.

For example, if mother wants to remind the child things to do in the morning:

Without loop:

Day 1:

1. Brush teeth
2. Wash face
3. Eat breakfast

Day 2:

1. Brush teeth
2. Wash face
3. Eat breakfast

Day 3:

1. Brush teeth
2. Wash face
3. Eat breakfast

Day 4:

1. Brush teeth
2. Wash face
3. Eat breakfast

With loop:

Everyday:

1. **Brush teeth**
2. **Wash face**
3. **Eat breakfast**

Loop

CUHK Jockey Club AI for the Future Project
中大賽馬會智為未來計劃

In this workshop we will only focus on 1 type of loops i.e., **for-loop**. **For-loop** is particularly useful for iteration over a list.

The syntax of for-loop is as follows:

```
for <variable> in <list> :
        Code that needs to be repeated
```

Loop

CUHK Jockey Club AI for the Future Project
中大賽馬會智為未來計劃

Let's try these exercises on Datalore to get familiar with the concept of **Loop**.

1. Create a loop that prints number 0-10.

2. Create a loop that prints "Hello" 5 times.

Function

CUHK Jockey Club AI for the Future Project
中大賽馬會智為未來計劃

Function is a block of organized, reusable code that is used to perform a single, related action.

You can pass parameters (optional) into a function.

The function can also return values (optional).

Define a function:

```
def Function_Name(parameter1, parameter2):
        Code block to run
```

Call a function:

```
Function_Name(parameter1_value, parameter2_value)
```

Function

CUHK Jockey Club AI for the Future Project
中大賽馬會智為未來計劃

Without using function:

Eat 

Peel the **apple** skin
Cut **apple** into pieces
Put **apple** into a bowl
Eat
Wash the bowl

Eat 

Peel the **pear** skin
Cut **pear** into pieces
Put **pear** into a bowl
Eat
Wash the bowl

Eat 

Peel the **peach** skin
Cut **peach** into pieces
Put **peach** into a bowl
Eat
Wash the bowl

Using function:

def **eat_fruit_function**(**fruit**):
Peel the **fruit** skin
Cut the **fruit** into pieces
Put the **fruit** into a bowl
Eat
Wash the bowl

eat_fruit_function(**apple**)

eat_fruit_function(**pear**)

eat_fruit_function(**peach**)

Function

- Mother: Eat the fruit and let me know if it is sweet

Define a function:

```
def eat_fruit_function(fruit):
    Peel the fruit skin
    Cut the fruit into pieces
    Put the fruit into a bowl
    Eat
    Judge how sweet it is
    Wash the bowl
    Return sweetness
```

Call a function:

eat_fruit_function(apple)

eat_fruit_function(pear)

eat_fruit_function(peach)

Return value:

Not sweet at all

Just so so

Sweet!

Function

CUHK Jockey Club AI for the Future Project
中大賽馬會智為未來計劃

Let's try these exercises on Datalore to get familiar with the concept of **Function.**

1. Create a function that prints "Welcome to the AI with Python workshop"
2. Create a function that takes two numbers as its parameters and returns the multiplication of those two numbers

Libraries

CUHK Jockey Club AI for the Future Project
中大賽馬會智為未來計劃

Libraries are a set of predefined code that can be re-used. It saves our time so that we don't have to code everything from scratch. **Libraries make coding a lot easier!**

Import whole library:

```
import library_name
```

Import a function from library:

```
from library_name import function_name
```

Libraries

CUHK Jockey Club AI for the Future Project
中大賽馬會智為未來計劃

Let's try these exercises on Datalore to get familiar with the concept of **Libraries.**

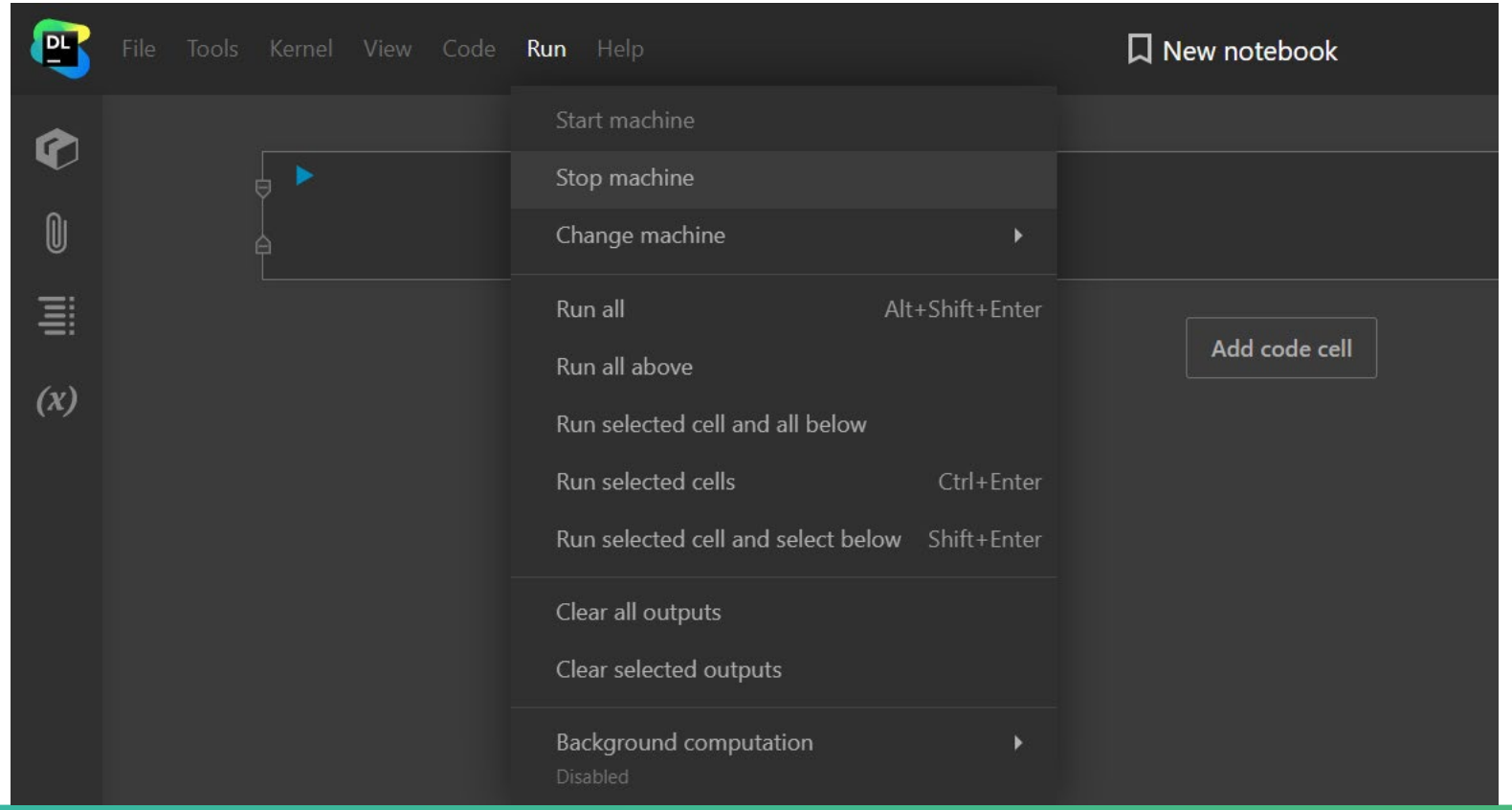
1. Install **art** library
2. Import **art** and try its **tprint** function

Closing Jupyter Notebook

CUHK Jockey Club AI for the Future Project
中大賽馬會智為未來計劃

1. Go to 'Run' and then press 'Stop machine'

2. Press the DL to close the notebook



Content

CUHK Jockey Club AI for the Future Project
中大賽馬會智為未來計劃

Getting Started

1. What is Python?

Python Basics

1. Hello World
2. Programming concepts (Variables, If-else statements, Loops, Lists, Dictionaries, Functions, and Libraries)

AI Project

1. Create a virtual assistant (tell joke, get weather, get movie rating, get movie director, get movie cast)

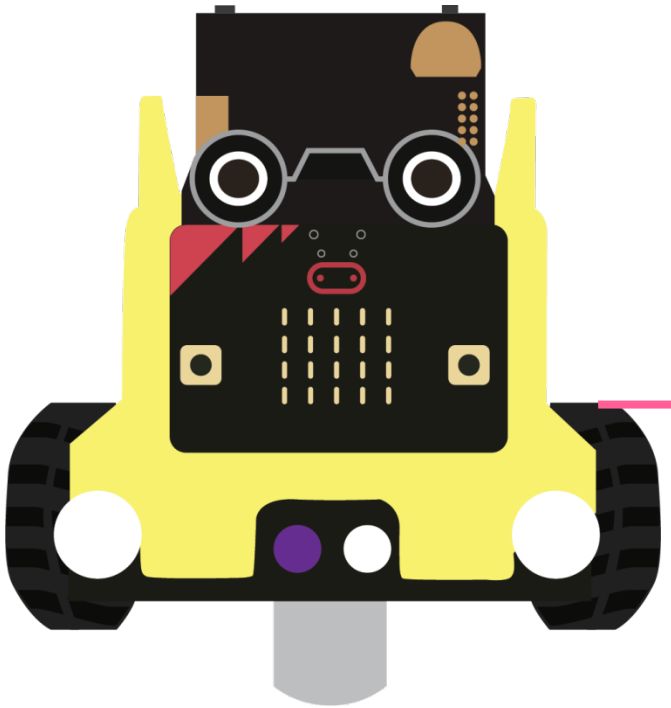
14:00 ~ 15:30
Lecture & Jupyter notebook 1

15:30 ~ 15:45
Breakout room/Take break

15:45 ~ 17:15
Lecture & Jupyter notebook 2
Survey

17:15 ~ 17:30
Breakout room/Early leave

CUHK Jockey Club AI for the Future Project
中大賽馬會智為未來計劃



AI Project

Content

Getting Started

1. What is Python?

Python Basics

1. Hello World
2. Programming concepts (Variables, If-else statements, Loops, Lists, Dictionaries, Functions, and Libraries)

AI Project

1. Create a virtual assistant (tell joke, get weather, get movie rating, get movie director, get movie cast)

14:00 ~ 15:30
Lecture & Jupyter notebook 1

15:30 ~ 15:45
Breakout room/Take break

15:45 ~ 17:15
Lecture & Jupyter notebook 2
Survey

17:15 ~ 17:30
Breakout room/Early leave

某程式師退休後決定練習書法，於是重金購買文房四寶。一日，飯後突生雅興，一番研墨擬紙，並點上上好檀香。定神片刻，潑墨揮毫，鄭重地寫下一行字：

Group Photo

What will we create?

CUHK Jockey Club AI for the Future Project
中大賽馬會智為未來計劃

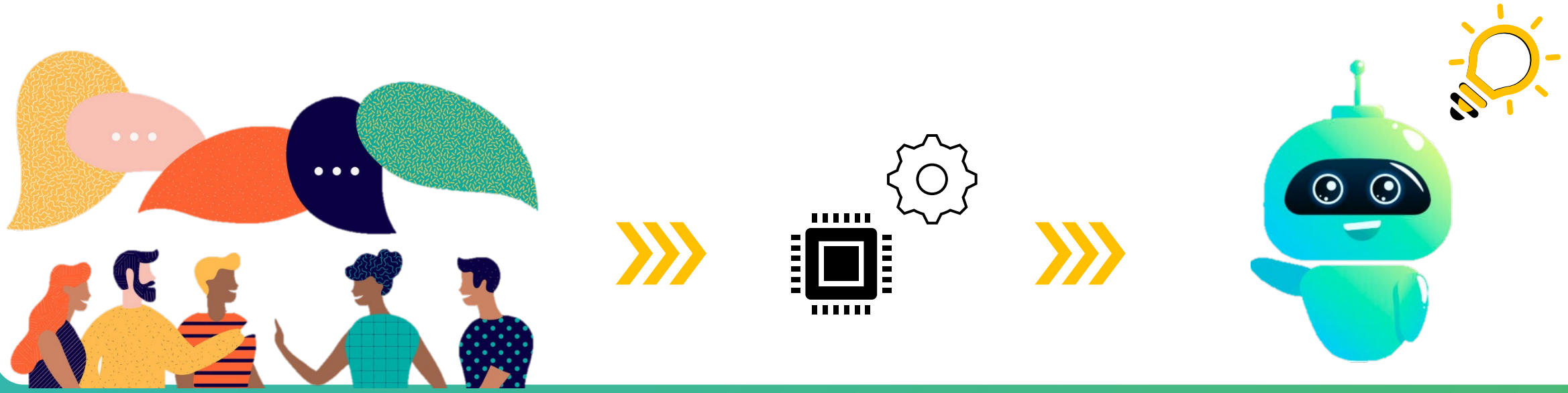
In this workshop, we will create a virtual assistant that understands English and can perform the following tasks –

1. Tell jokes
2. Get current weather of a city
3. Get movie ratings
4. Find the director(s) name of a movie
5. Find the cast of a movie

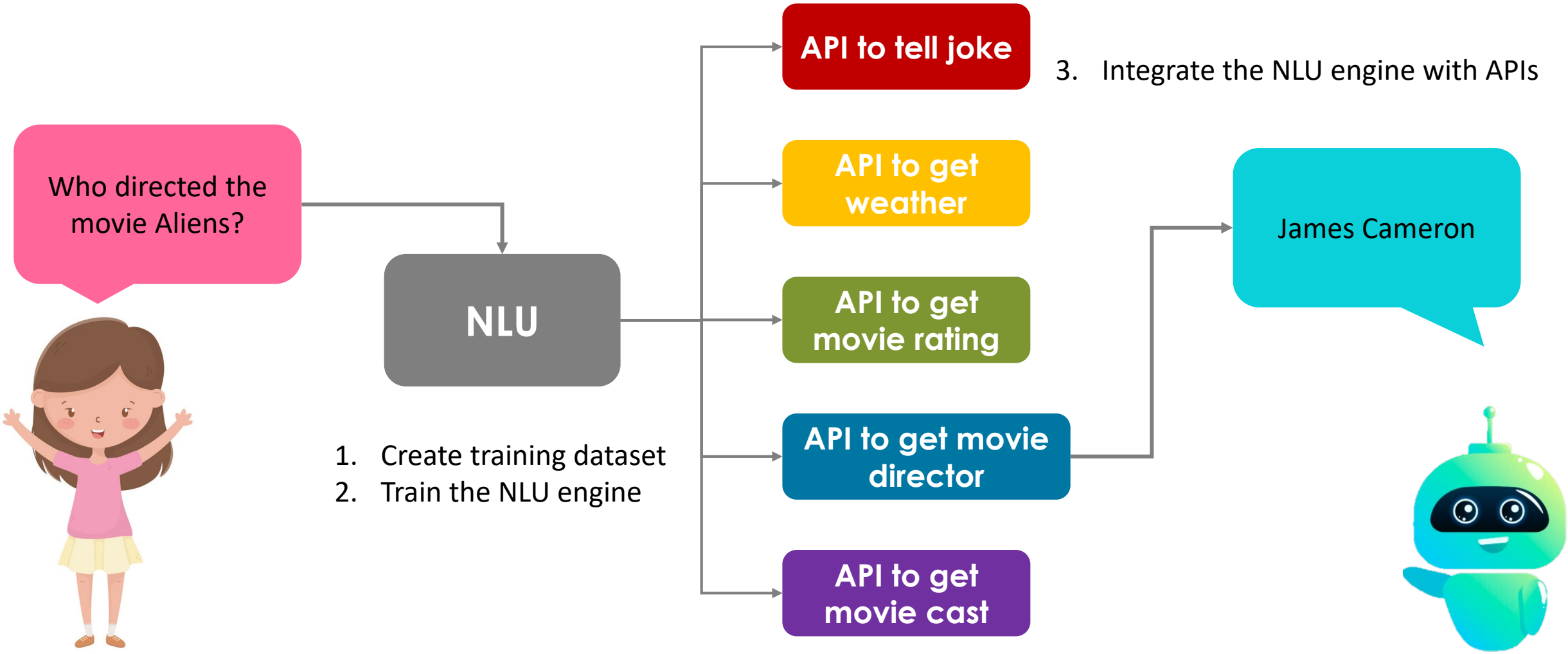
Natural Language Understanding

CUHK Jockey Club AI for the Future Project
中大賽馬會智為未來計劃

Natural Language Understanding (NLU) is a subtopic in Natural Language Processing (NLP) that deals with **transforming human language to something that a machine can understand**.



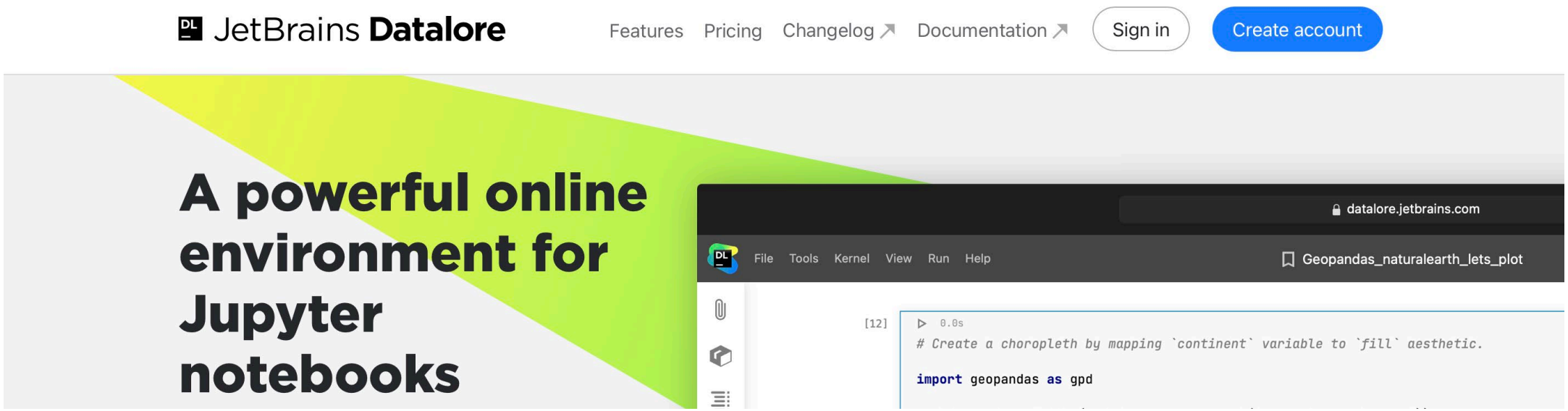
AI Project



Open JetBrains Datalore

CUHK Jockey Club AI for the Future Project
中大賽馬會智為未來計劃

Access and sign in to JetBrains **Datalore** Platform using the following link: datalore.jetbrains.com

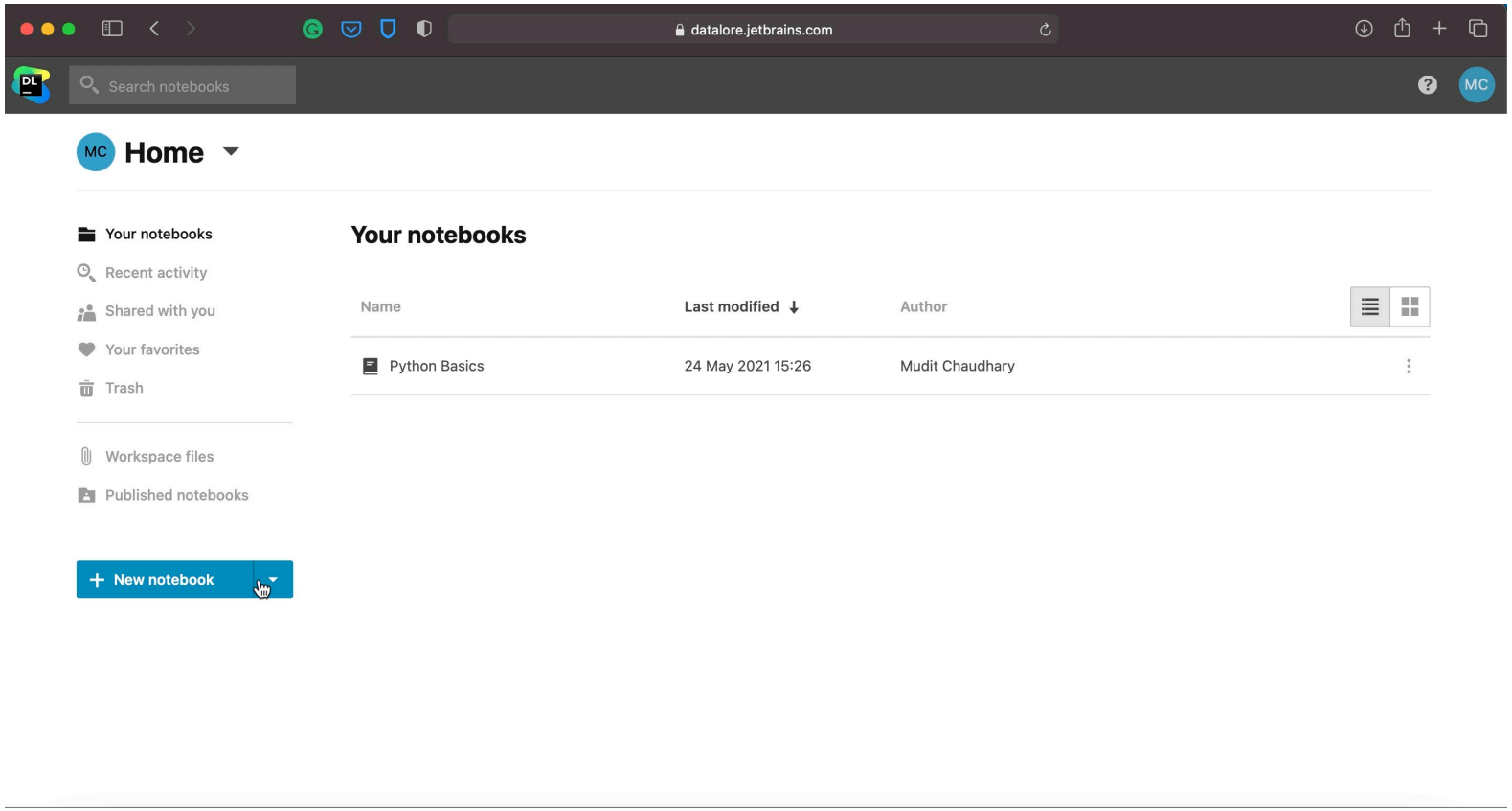


Open JetBrains Datalore

CUHK Jockey Club AI for the Future Project
中大賽馬會智為未來計劃

1. Open our tutorial notebook by pressing the arrow beside 'New Notebook' and select 'Upload notebook'

2. Go to your browser and go to cutt.ly/rHqLKSS in the URL bar. Copy the full URL into DataLore and press 'Upload' button



Create Training Dataset

CUHK Jockey Club AI for the Future Project
中大賽馬會智為未來計劃

In this workshop, we will learn the following concepts of NLU that are used in chatbots, virtual assistants and dialog systems:

- Utterances
- Intents
- Slots
- Entities

Create Training Dataset

intent slots utterances entity

```
# get_weather intent
---
type: intent
name: get_weather
slots:
  - name: city
    entity: city

utterances:
  - How is the weather in [city](London)?
  - What is the weather in [city](New York)?
  - Can you tell me how is the weather like in [city](Hong Kong)?
  - I wonder how hot is it in [city](Paris)?
```

```
# city entity
---
type: entity
name: city
values:
  - Hong Kong
  - New York
  - Paris
  - London
  - Tokyo
  - Shanghai
```

Utterances

intent slots utterances entity

```
# get_weather intent
---
type: intent
name: get_weather
slots:
  - name: city
    entity: city

utterances:
  - How is the weather in [city](London)?
  - What is the weather in [city](New York)?
  - Can you tell me how is the weather like in [city](Hong Kong)?
  - I wonder how hot is it in [city](Paris)?
```

```
# city entity
---
type: entity
name: city
values:
  - Hong Kong
  - New York
  - Paris
  - London
  - Tokyo
  - Shanghai
```

Utterances

- Anything that a user says is an **utterance**
- The virtual assistant uses sample **utterances** as training data to construct a model to detect **intents** and **entities**

E.g.,

- Can you tell me how the weather in Hong Kong is?
- I'm bored! Tell me a joke.
- What is the cast for the movie the matrix?

Intent

intent

slots

utterances

```
# get_weather intent
---
type: intent
name: get_weather
slots:
  - name: city
    entity: city

utterances:
  - How is the weather in [city](London)?
  - What is the weather in [city](New York)?
  - Can you tell me how is the weather like in [city](Hong Kong)?
  - I wonder how hot is it in [city](Paris)?
```

entity

```
# city entity
---
type: entity
name: city
values:
  - Hong Kong
  - New York
  - Paris
  - London
  - Tokyo
  - Shanghai
```

Intent

CUHK Jockey Club AI for the Future Project
中大賽馬會智為未來計劃

Intent defines the action or intention contained in the utterance. It tells the computer what we ask it to do.

Can you guess what the intent of the following utterances is?

1. Can you tell me how the weather in Hong Kong is?

   Intent: Get the weather of a city

2. I'm bored! Tell me a joke.

   Intent: Tell a joke

3. Turn on the light in the kitchen.

   Intent: Switch light on

Slots

intent

slots

utterances

entity

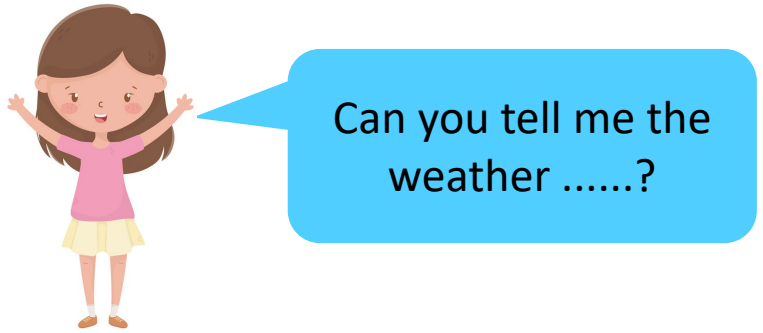
```
# get_weather intent
---
type: intent
name: get_weather
slots:
  - name: city
    entity: city

utterances:
  - How is the weather in [city](London)?
  - What is the weather in [city](New York)?
  - Can you tell me how is the weather like in [city](Hong Kong)?
  - I wonder how hot is it in [city](Paris)?
```

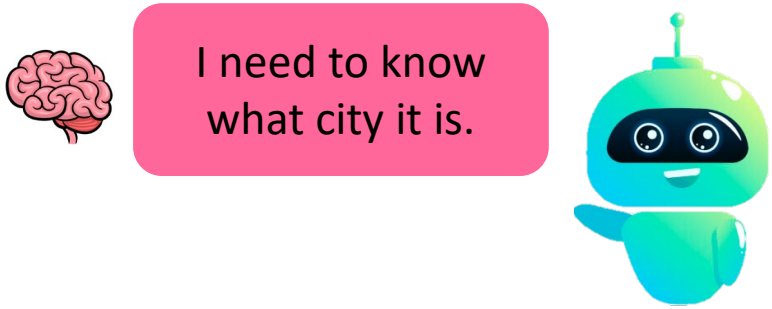
```
# city entity
---
type: entity
name: city
values:
  - Hong Kong
  - New York
  - Paris
  - London
  - Tokyo
  - Shanghai
```

Slots

If I ask you to tell me the current weather of a city? What type of information do you think is required?

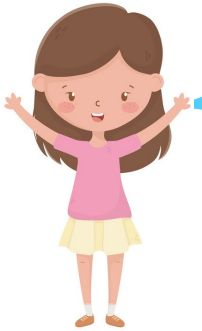


The virtual assistant needs to know what the **City** is. So, here the required information is **City**.



Slots

Once the intent is known, we extract some additional information that is required to fulfil the intent. This **additional information** is called a **slot.**



Can you tell me the weather of Taipei?

The current temperature in Taipei is 29 degrees Celsius.

You can think of intent as a function and slots as its parameters.

Slots

CUHK Jockey Club AI for the Future Project
中大賽馬會智為未來計劃

Can you guess what slot needs to be filled to fulfil the intent?

1. Can you tell me how the weather in Hong Kong is?

   Intent: Get the weather of a city

   Required Slot: City

2. How good is the movie Batman?

   Intent: Get the rating of a movie

   Required Slot: Movie name

3. Show me flights from Hong Kong to Shanghai leaving on July 6th.

   Intent: Search flight

   Required Slot: Origin city, Destination city, Date of travel

Entity

intent slots utterances

```
# get_weather intent
---
type: intent
name: get_weather
slots:
  - name: city
    entity: city

utterances:
  - How is the weather in [city](London)?
  - What is the weather in [city](New York)?
  - Can you tell me how is the weather like in [city](Hong Kong)?
  - I wonder how hot is it in [city](Paris)?
```

entity

```
# city entity
---
type: entity
name: city
values:
  - Hong Kong
  - New York
  - Paris
  - London
  - Tokyo
  - Shanghai
```

Entity

CUHK Jockey Club AI for the Future Project
中大賽馬會智為未來計劃

Entity (or Slot Type) describe the type of a slot value. It is like data type in programming.

Examples of entity and its values:

Entity: City

Values: Hong Kong, New York, London, Paris, etc.

Entity: Date

Values: 12-January-2020, 19/01/1999, 12-June-2020, etc.

Entity: Movie

Values: Star Wars, Spider-man, Titanic, etc.

Entity

CUHK Jockey Club AI for the Future Project
中大賽馬會智為未來計劃

Different **slots** can be of the same **entity (or slot type)**.

1. Show me flights from Hong Kong to Shanghai leaving on July 6th.

   Intent: Search flight

   Slots: Origin city, Destination city, Date of travel

```
# search_flight intent
---
type: intent
name: searchFlight
slots:
  - name: origin
    entity: city
  - name: destination
    entity: city
  - name: date
    entity: snips/datetime
utterances:
  - Find me a flight from [origin](Hong Kong) to [destination](New York)
  - I need a flight leaving [date](this weekend) from [origin](Hong Kong) to [destination](New York)
  - Show me flights from [origin](Hong Kong) to [destination](New York) leaving [date](this evening)
```

```
# city entity
---
type: entity
name: city
values:
  - Hong Kong
  - New York
  - Paris
  - London
  - Tokyo
  - Shanghai
```

Create Training Dataset

CUHK Jockey Club AI for the Future Project
中大賽馬會智為未來計劃

The **intents** we need to create are:

- **tell_joke**: To detect that the user is asking the virtual assistant for a joke. There are no slots required for this intent.
- **get_weather**: To detect that the user is asking for current weather of a city. For this intent we need to fill a slot for **city**.
- **get_rating**: To detect that the user is asking the rating of a movie. For this intent we need to fill a slot for **movie_name**.
- **get_director**: To detect that the user is asking for who is the director of a movie. For this intent we need to fill a slot for **movie_name**.
- **get_cast**: To detect that the user is asking for who acted in a movie. For this intent we need to fill a slot for **movie_name**.

The **entities** we need to create are: **city**, **movie_name**

AI Project

Who directed the movie Aliens?

NLU

1. Create training dataset
2. Train the NLU engine

Now What?

CUHK Jockey Club AI for the Future Project
中大賽馬會智為未來計劃

We learned about Utterances, Intents, Slots and Entities.

But…

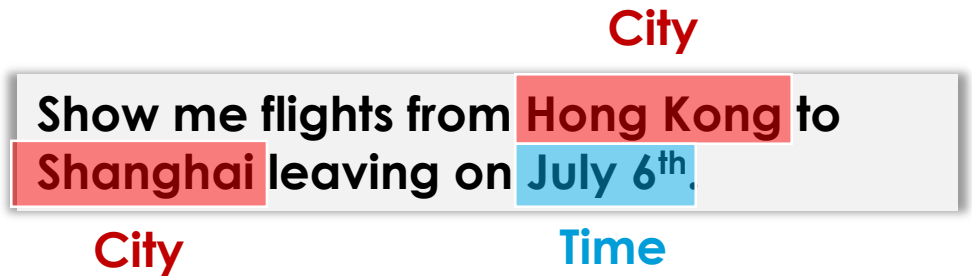
What is AI?

What is the role of AI here?

Why aren't we using AI?

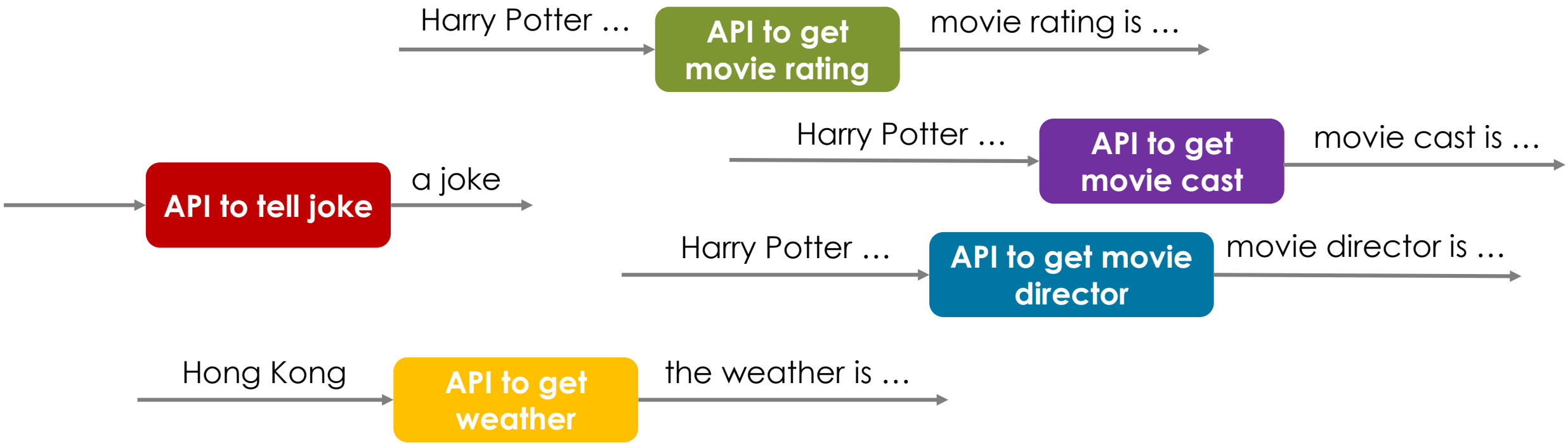
AI's Role

- We need AI to understand the utterance in natural language and automatically –
  - Classify the underlying intent of the utterance
  - Extract the entities that may be contained in the utterance

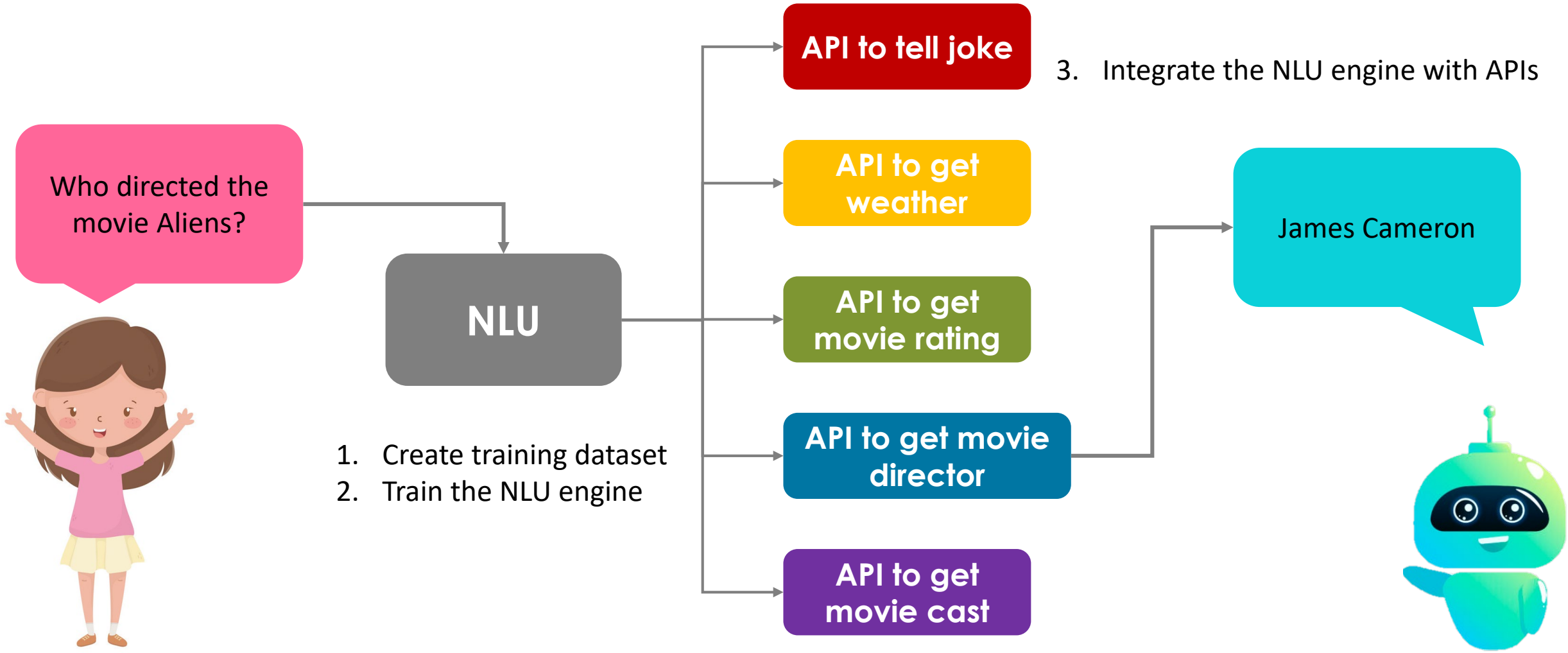


API

- Application programming interface (API), which is a set of definitions and protocols for building and integrating application software.



AI Project



Further Readings

xingxing@cuhk.edu.hk

- https://snips-nlu.readthedocs.io/en/latest/tutorial.html#tutorial
- https://rasa.com/blog/rasa-nlu-in-depth-part-1-intent-classification/
- https://rasa.com/blog/rasa-nlu-in-depth-part-2-entity-recognition/
- https://botfront.io/blog/how-intent-classification-works-in-nlu
- https://www.analyticsvidhya.com/blog/2021/06/part-10-step-by-step-guide-to-master-nlp-named-entity-recognition/
- Jinjie Ni, Tom Young, Vlad Pandelea, Fuzhao Xue, Vinay Adiga, and Erik Cambria. 2021. Recent advances in deep learning based dialogue systems: A systematic survey.
- https://ai.googleblog.com/2020/01/towards-conversational-agent-that-can.html
- https://ai.facebook.com/blog/blender-bot-2-an-open-source-chatbot-that-builds-long-term-memory-and-searches-the-internet/

Future Teacher Training Workshops

CUHK Jockey Club AI for the Future Project
中大賽馬會智為未來計劃

- Hands-on Machine Learning Crash Course:

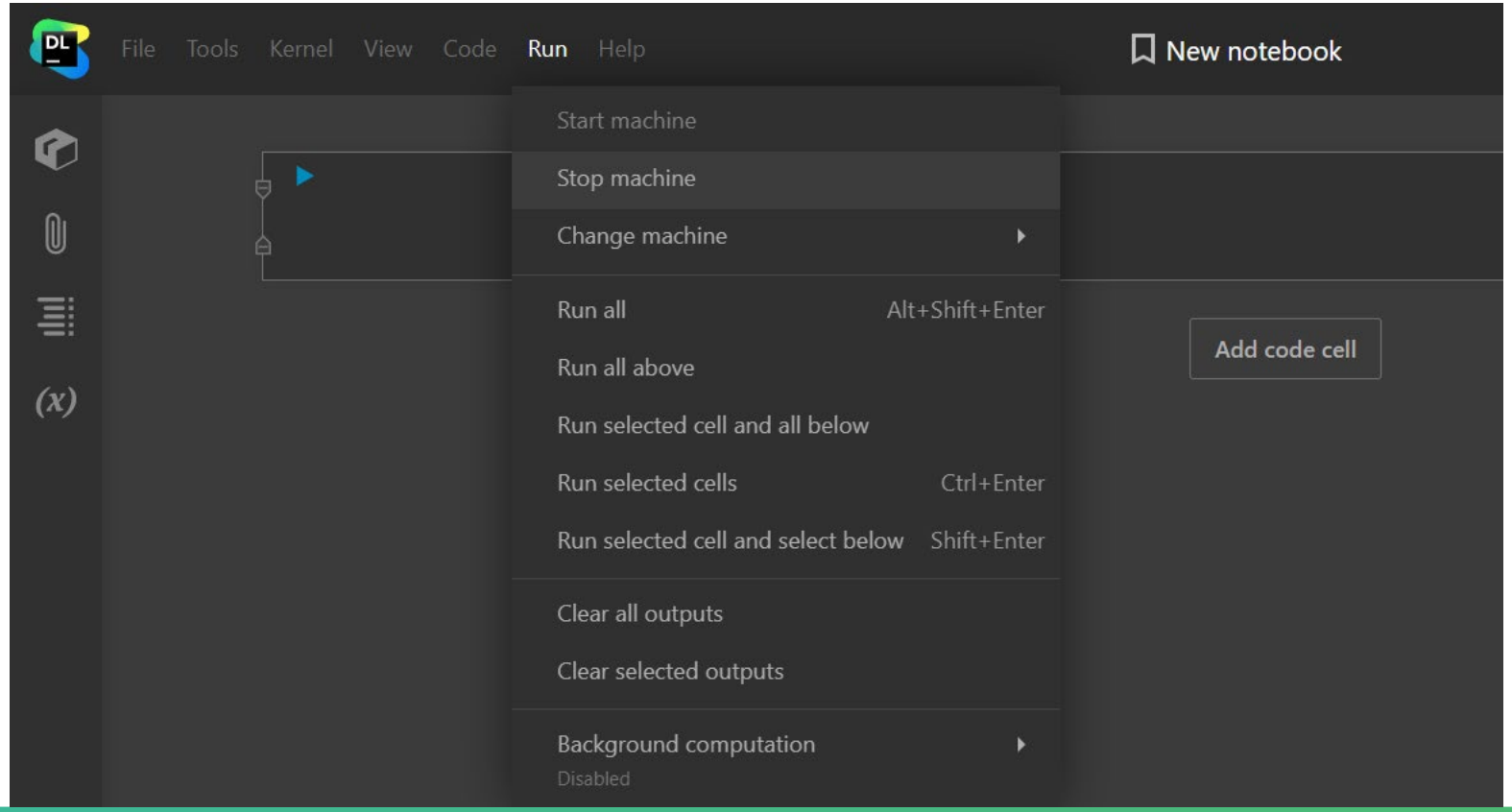
	Concept	Hands-on project	Chapter	Details
1	Machine learning basics	Algorithm	2: Fundamentals of AI	Forward pass: data split, regression, classification, loss, accuracy, Backward pass: gradient decent, learning rate, optimization, overfitting
2	Convolutional Neural Network (CNN)	Image classification	3: See	Neural network, layer, weight, tensor, neuron, fully-connected layer, convolutional layer, activation function, pooling layer, flatten layer
3	Reinforcement learning (RL)	3D VR game	8: Simulation	5 key concepts (agent, environment, action, observation, reward/punishment), deep reinforcement learning
4	Recurrent Neural Network (RNN)	Write a book	6: Read	Recurrent layer, types of RNN (one-to-one, one-to-many, many-to-one, many-to-many), word segmentation, word embedding
5	Long Short-term Memory (LSTM)	Time series forecast	7: AI reasoning	Exploding gradient problem, vanishing gradient problem, the gates of LSTM (forget gate, input gate, relevance gate, output gate), principle component analysis (PCA)
6	Generative adversarial network (GAN)	...	…	…

Closing Jupyter Notebook

1. End virtual assistant by typing “Bye” or “bye”

2. Go to ‘Run’ and then press ‘Stop machine’

3. Press the DL to close the notebook



AI for the Future
CUHK Jockey Club AI for the Future Project
中大賽馬會智為未來計劃

聯合主辦 Co-organized by:
The Chinese University of Hong Kong
Faculty of Engineering
Faculty of Education

捐助機構 Funded by:
香港賽馬會慈善信託基金
The Hong Kong Jockey Club Charities Trust

Python for AI Workshop Evaluation Survey (May 2022)

Please scan the QR code to fill in the survey

Or click the link
https://forms.gle/esRFW8qopGf3WS6M7

Content

CUHK Jockey Club AI for the Future Project
中大賽馬會智為未來計劃

Getting Started

1. What is Python?

Python Basics

1. Hello World
2. Programming concepts (Variables, If-else statements, Loops, Lists, Dictionaries, Functions, and Libraries)

AI Project

1. Create a virtual assistant (tell joke, get weather, get movie rating, get movie director, get movie cast)

14:00 ~ 15:30
Lecture & Jupyter notebook 1

15:30 ~ 15:45
Breakout room/Take break

15:45 ~ 17:15
Lecture & Jupyter notebook 2
Survey

17:15 ~ 17:30
Breakout room/Early leave