

## Lesson 01 Demo 05

### Storing User Information Using Different Data Types

**Objective:** To demonstrate how Python handles different data types, focusing on storing and displaying user input to build a strong foundation for data processing

**Tools required:** Anaconda Navigator (Jupyter Notebook)

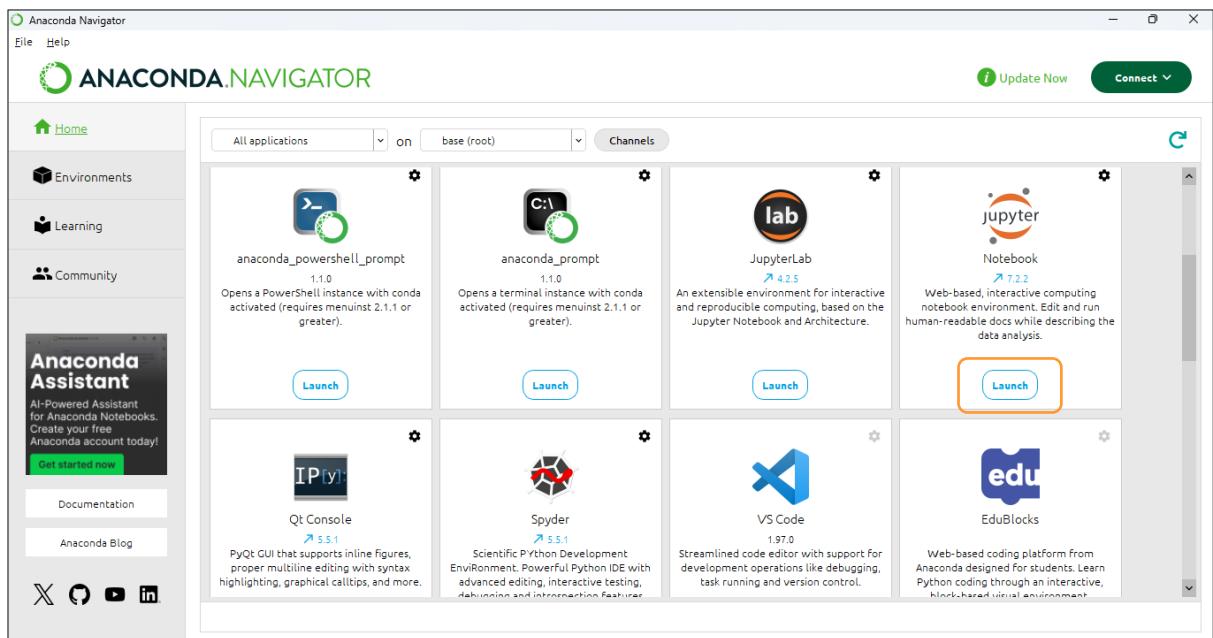
**Prerequisites:** Basic understanding of Python syntax

**Steps to be followed:**

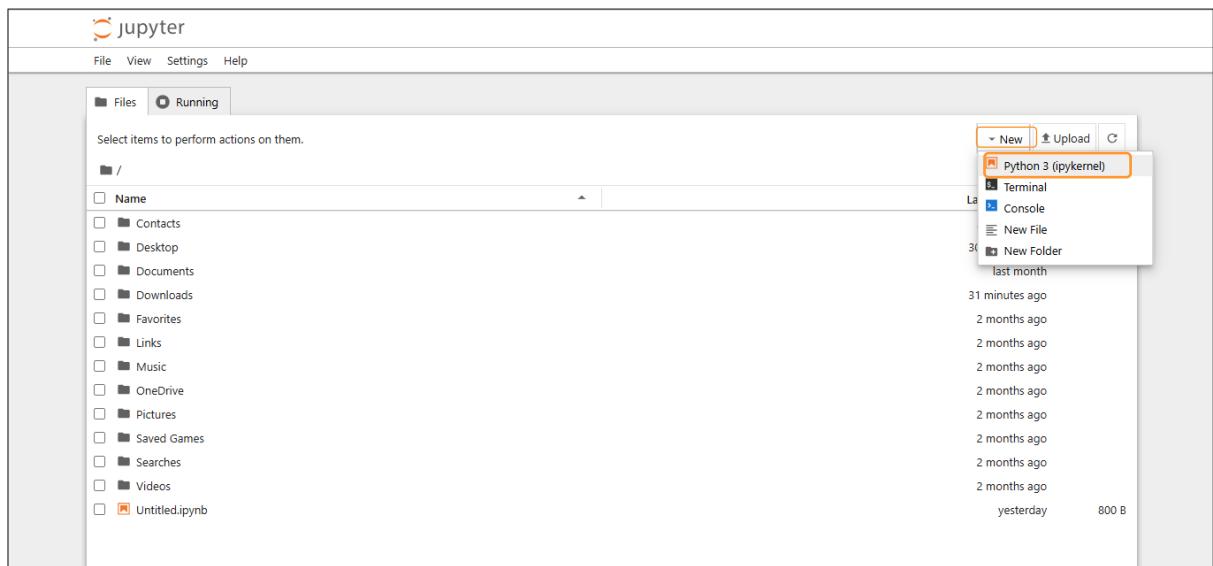
1. Set up Jupyter Notebook and create a Python file
2. Write and run the code

#### Step 1: Set up Jupyter Notebook and create a Python file

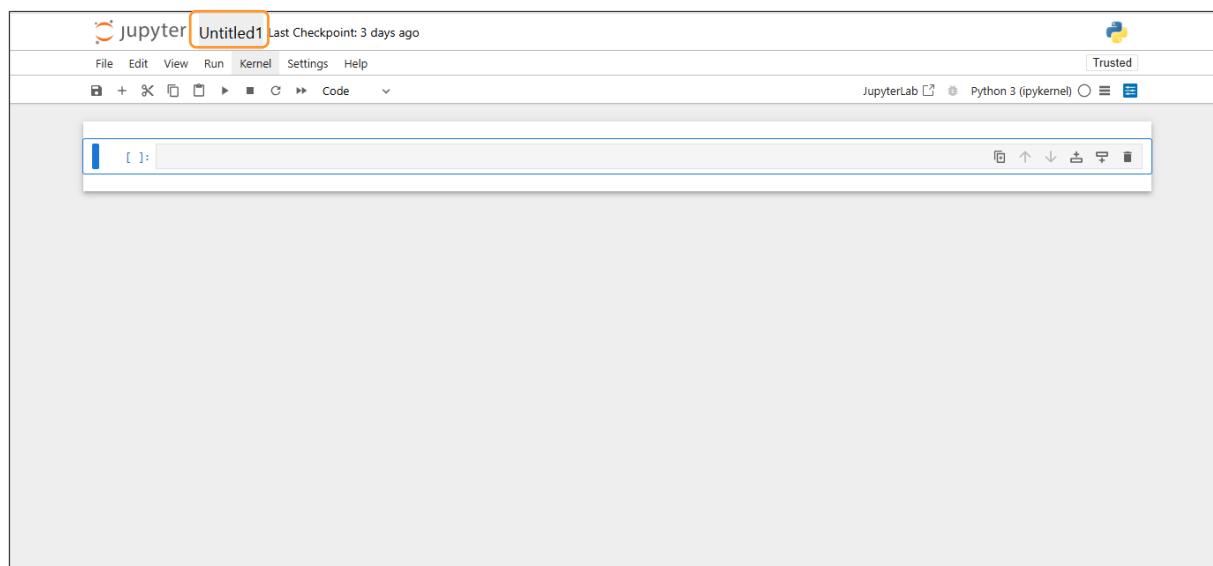
1.1 Open Anaconda Navigator and click on **Launch** below Jupyter Notebook



1.2 Once the interface opens, click on **New** and then on **Python 3 (ipykernel)** to create a new notebook



1.3 Click on the title (**Untitled1**) and rename the notebook to **User\_Data\_Types.ipynb**

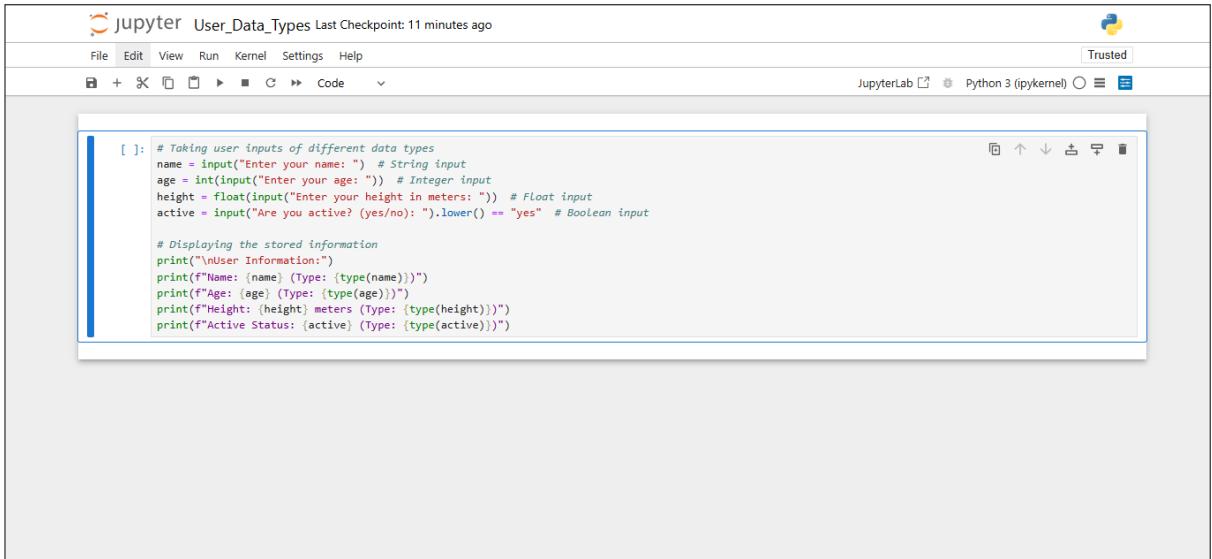


## Step 2: Write and run the code

2.1 In the first code cell of the notebook, write the following script:

```
# Taking user inputs of different data types
name = input("Enter your name: ") # String input
age = int(input("Enter your age: ")) # Integer input
height = float(input("Enter your height in meters: ")) # Float input
active = input("Are you active? (yes/no): ").lower() == "yes" # Boolean input

# Displaying the stored information
print("\nUser Information:")
print(f"Name: {name} (Type: {type(name)} )")
print(f"Age: {age} (Type: {type(age)} )")
print(f"Height: {height} meters (Type: {type(height)} )")
print(f"Active Status: {active} (Type: {type(active)} )")
```

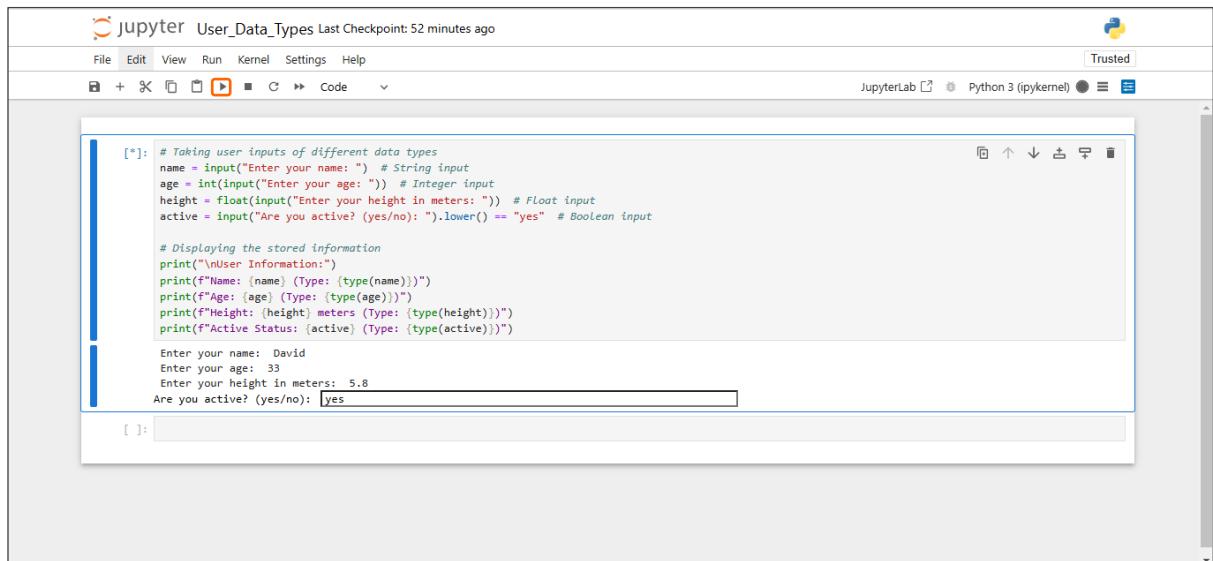


The screenshot shows a Jupyter Notebook window titled 'jupyter User\_Data\_Types Last Checkpoint: 11 minutes ago'. The window has a 'Trusted' status indicator at the top right. Below the title bar is a menu bar with 'File', 'Edit', 'View', 'Run', 'Kernel', 'Settings', and 'Help'. To the right of the menu bar are icons for 'JupyterLab', 'Python 3 (ipykernel)', and other notebook-related functions. The main area contains a code cell with the following Python code:

```
[ ]: # Taking user inputs of different data types
name = input("Enter your name: ") # String input
age = int(input("Enter your age: ")) # Integer input
height = float(input("Enter your height in meters: ")) # Float input
active = input("Are you active? (yes/no): ").lower() == "yes" # Boolean input

# Displaying the stored information
print("\nUser Information:")
print(f"Name: {name} (Type: {type(name)} )")
print(f"Age: {age} (Type: {type(age)} )")
print(f"Height: {height} meters (Type: {type(height)} )")
print(f"Active Status: {active} (Type: {type(active)} )")
```

2.2 Click the **Play (▶)** button (or press Shift and Enter on your keyboard) to execute the script. Enter the required details when prompted.



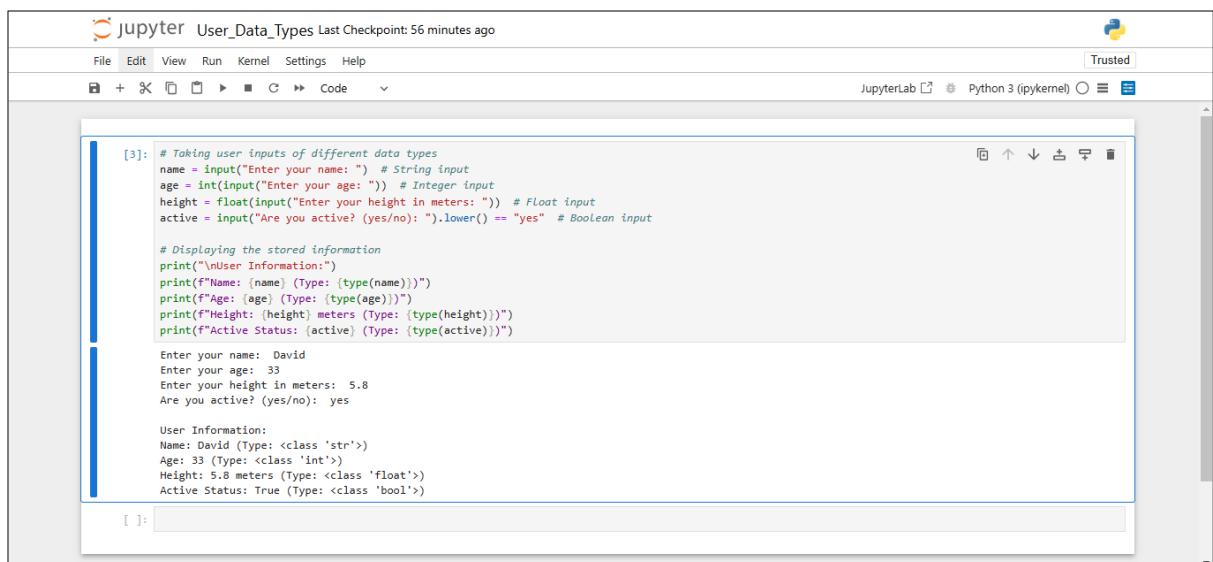
A screenshot of a Jupyter Notebook interface. The title bar says "jupyter User\_Data\_Types Last Checkpoint: 52 minutes ago". The menu bar includes File, Edit, View, Run, Kernel, Settings, Help, and a Trusted badge. Below the menu is a toolbar with icons for file operations and a play button. The code cell contains Python code for taking user inputs and printing them with their types. The output cell shows the user's input followed by the printed information.

```
[*]: # Taking user inputs of different data types
name = input("Enter your name: ") # String input
age = int(input("Enter your age: ")) # Integer input
height = float(input("Enter your height in meters: ")) # Float input
active = input("Are you active? (yes/no): ").lower() == "yes" # Boolean input

# Displaying the stored information
print("\nUser Information:")
print(f"Name: {name} (Type: {type(name)}")
print(f"Age: {age} (Type: {type(age)}")
print(f"Height: {height} meters (Type: {type(height)}")
print(f"Active Status: {active} (Type: {type(active)})")

Enter your name: David
Enter your age: 33
Enter your height in meters: 5.8
Are you active? (yes/no): yes
```

2.3 Observe the output displaying the values and their respective data types



A screenshot of a Jupyter Notebook interface. The title bar says "jupyter User\_Data\_Types Last Checkpoint: 56 minutes ago". The menu bar includes File, Edit, View, Run, Kernel, Settings, Help, and a Trusted badge. Below the menu is a toolbar with icons for file operations and a play button. The code cell contains Python code for taking user inputs and printing them with their types. The output cell shows the user's input followed by the printed information, including the type of each variable.

```
[3]: # Taking user inputs of different data types
name = input("Enter your name: ") # String input
age = int(input("Enter your age: ")) # Integer input
height = float(input("Enter your height in meters: ")) # Float input
active = input("Are you active? (yes/no): ").lower() == "yes" # Boolean input

# Displaying the stored information
print("\nUser Information:")
print(f"Name: {name} (Type: {type(name)}")
print(f"Age: {age} (Type: {type(age)}")
print(f"Height: {height} meters (Type: {type(height)}")
print(f"Active Status: {active} (Type: {type(active)})")

Enter your name: David
Enter your age: 33
Enter your height in meters: 5.8
Are you active? (yes/no): yes

User Information:
Name: David (Type: <class 'str'>)
Age: 33 (Type: <class 'int'>)
Height: 5.8 meters (Type: <class 'float'>)
Active Status: True (Type: <class 'bool'>)
```

By following these steps, you have successfully explored how Python handles different data types using user input in Jupyter Notebook and how it processes and stores strings, integers, floats, and boolean values.