

ML Job Ready Course

Week 1 Day 1

Md. Mehedi Hasan

Senior Machine Learning Engineer

Minerva Analytics Inc

Conda command

1. How to create an environment in conda?

conda create --name myenv python=3.x --no-default-packages

2. How to activate env?

conda activate myenv

2.1: How to install pip?

conda install pip

3. How to install dependencies?

pip install -r requirements.txt

3.1 How to store packages?

conda list --export | cut -d= -f1,2 > requirements.txt

pip list --format=freeze > requirements.txt

4. To see how many Conda environments are available on your system, you can use the following command:

conda env list / conda list -n myenv

or alternatively:

conda info --envs

5. How to remove conda env?

Deactivate the current environment (if you're inside the environment you want to remove):

If you're in the environment you want to remove, you first need to deactivate it:

conda deactivate

6. Remove the environment:

Run the following command to remove a specific Conda environment. Replace myenv with the name of the environment you want to delete:

conda env remove --name myenv

This will delete the environment and all its associated packages.

Python Basic

1. Variables and Data Types

Variables are used to store data in memory. Common types are:

- *int* (e.g., 10)
- *float* (e.g., 3.14)
- *str* (e.g., 'hello')
- *bool* (e.g., True, False)

```
name = "Alice"
```

```
age = 30
```

```
height = 5.4
```

```
is_student = True
```

2. Type Casting

Convert one data type to another using `int()`, `float()`, `str()`, etc.

```
age = "25"
```

```
age_int = int(age)
```

```
print(age_int + 5)
```

3. `input()` and `print()` functions

Use `input()` to get user input and `print()` to display output.

```
name = input("Enter your name: ")
```

```
print("Hello, " + name + "!")
```

4. String Formatting

Combine strings and variables using f-strings.

```
name = "Bob"
```

```
age = 22
```

```
print(f"My name is {name} and I am {age} years old.")
```

5. Comments

Use `#` for single-line comments to explain your code.

```
# This is a comment
```

```
name = "John" # Storing the name
```

6. Indentation Rules

Python uses indentation to define blocks of code.

```
age = 18
```

```
if age >= 18:
```

```
    print("You are an adult.")
```

Practice: BMI Calculator

Program to take user input and calculate BMI.

```
name = input("Enter your name: ")
```

```
height = float(input("Enter your height in meters: "))
```

```
weight = float(input("Enter your weight in kilograms: "))
```

```
bmi = weight / (height ** 2)
```

```
print(f"{name}, your BMI is: {bmi:.2f}")
```

Practice: Personal Info Display

Print your name, age, and favorite number.

```
name = "John Doe"
```

```
age = 25
```

```
fav_number = 7
```

```
print(f"My name is {name}, I am {age} years old, and my favorite number is {fav_number}.")
```

Practice: Type Casting Demo

Demo program for type casting user input.

```
age = input("Enter your age: ")
```

```
age = int(age)
```

```
print(f"In 5 years, you will be {age + 5} years old.")
```

Practice: Comment and Indentation

Demo to show comments and proper indentation.

```
# This is a simple program
```

```
# It adds two numbers
```

```
num1 = 10
```

```
num2 = 20
```

```
# Add them
```

```
result = num1 + num2
```

```
# Print the result
```

```
print("The sum is:", result)
```