Preparation Phase
Prior Knowledge

Date	25 October 2022
Team Members	917719C051, 917719C069, 917719C079, 917719C079
Project Name	Project – Car Resale Value Prediction

Prior Knowledge – Car Resale Value Prediction:

1. **Supervised and Unsupervised learning:**
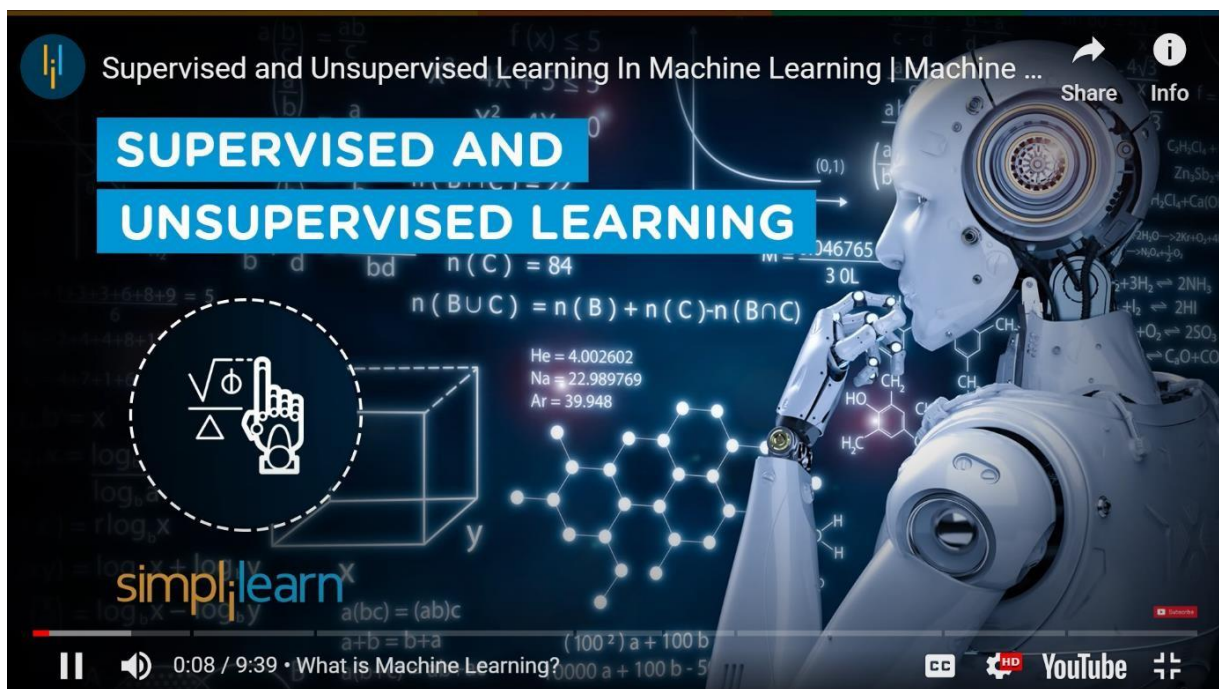
What is Machine Learning?

What is Machine Learning?

Machine Learning is the science of making computers learn and act like humans by feeding data and information without being explicitly programmed!
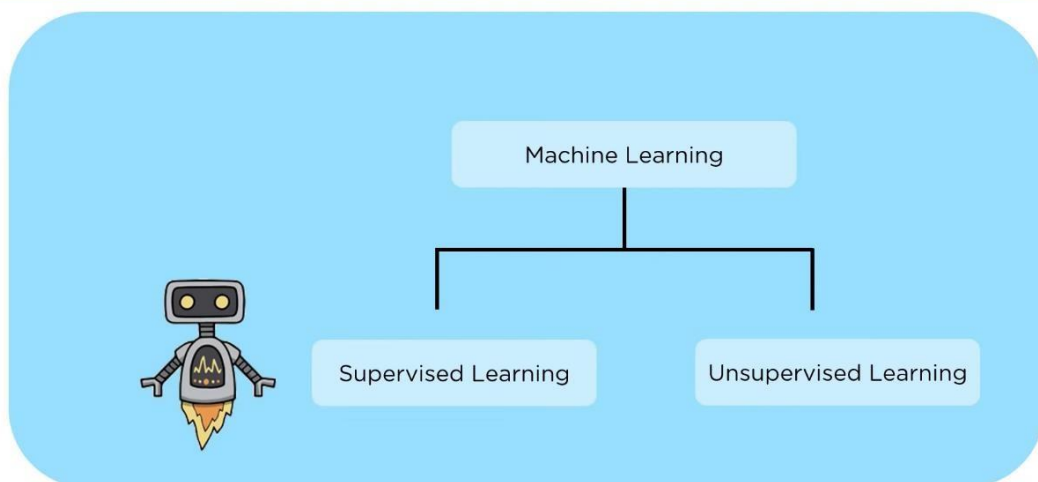
Past Data — Train → Analyse → System Learns — Prediction → Output

Data is processed

System Learns

Machine Learning makes predictions and decisions based on past data

Types of Machine Learning:

Types of Machine Learning

Machine Learning

Supervised Learning
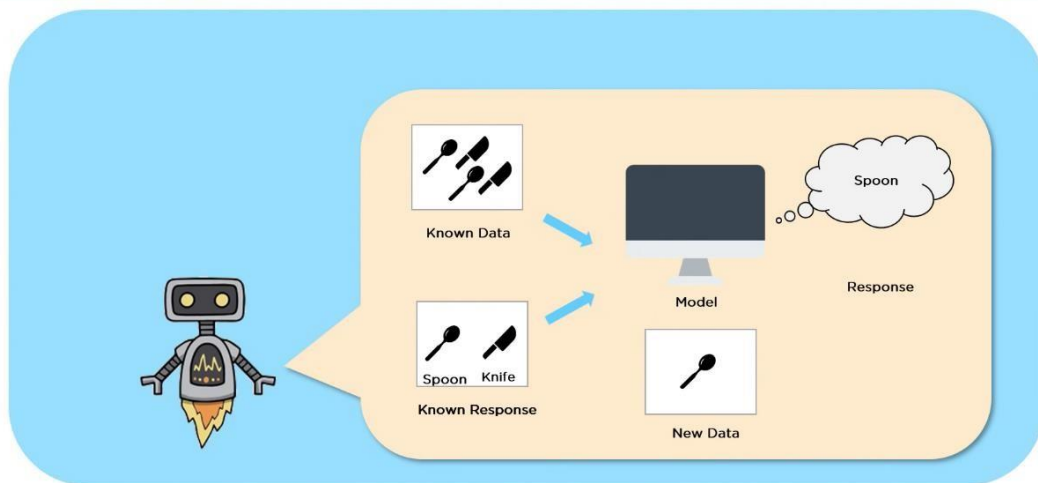
Unsupervised Learning

Supervised Learning:

Supervised Learning
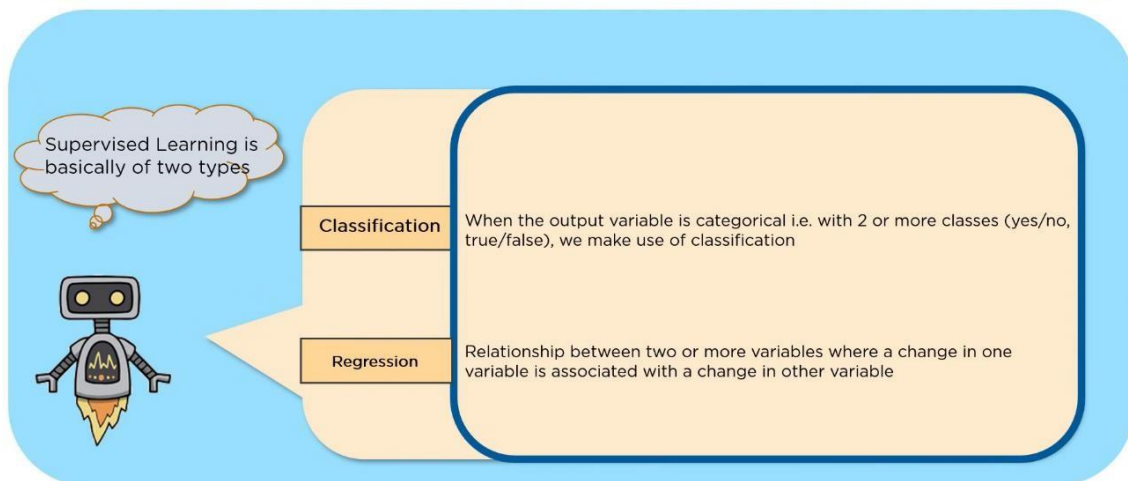
Types of Supervised Learning:

Types of Supervised Learning

Applications of Supervised Learning:
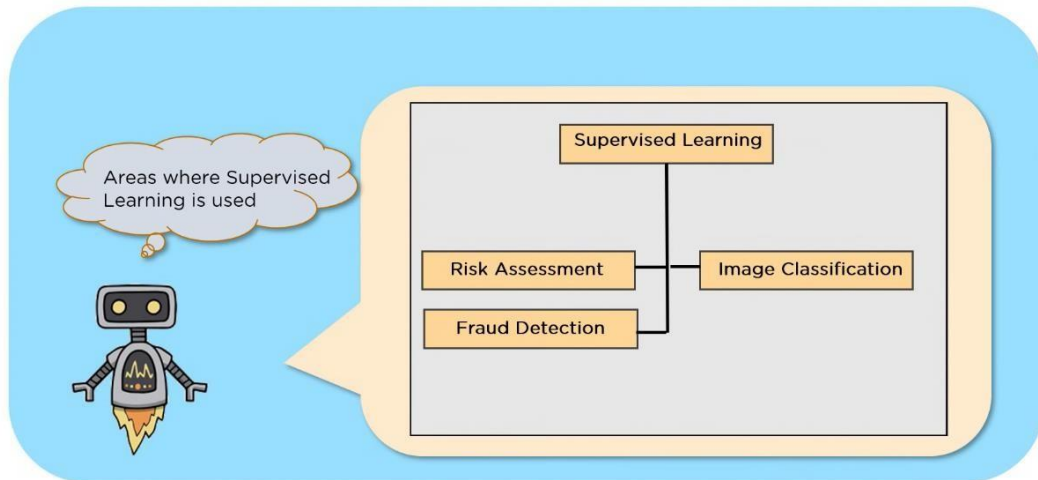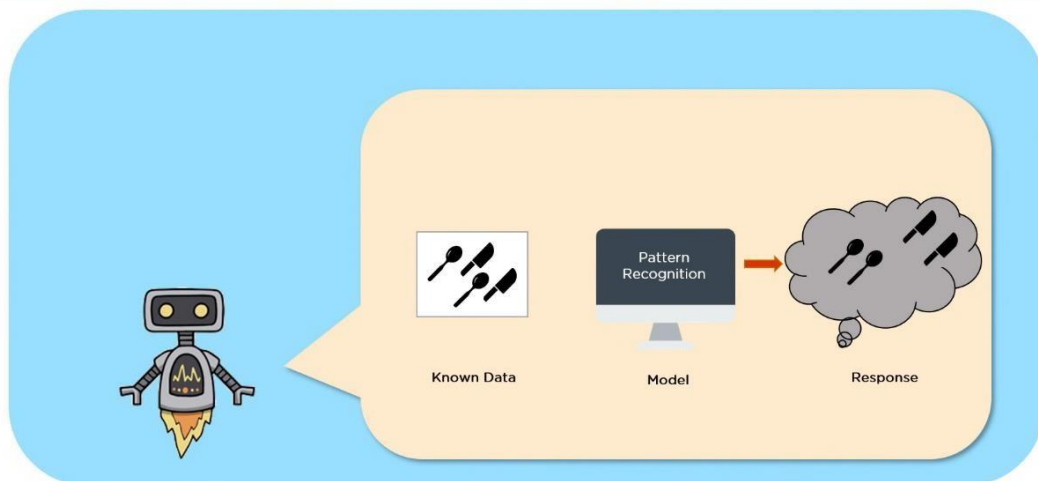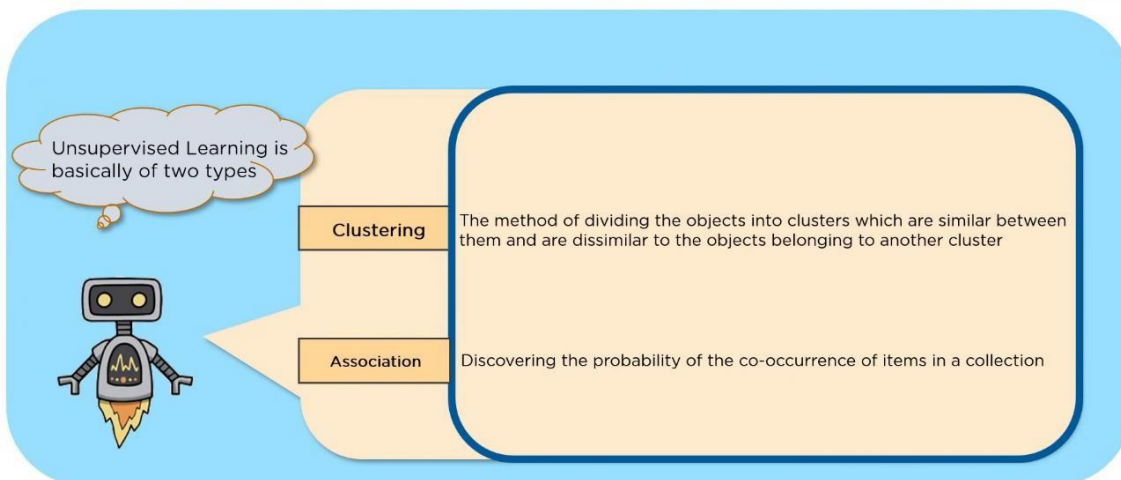
Unsupervised Learning:

Types of Unsupervised Learning:

Types of Unsupervised Learning

Applications of Unsupervised Learning:

Applications of Unsupervised Learning
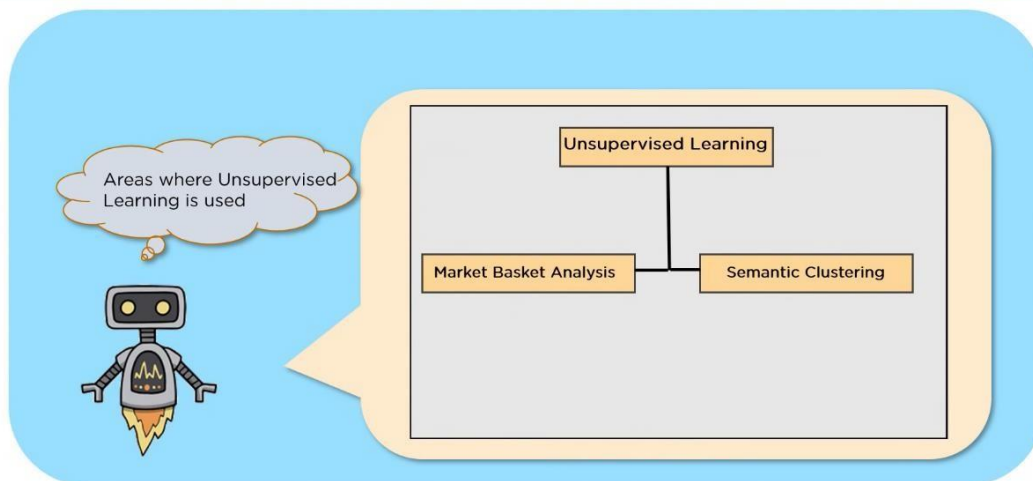
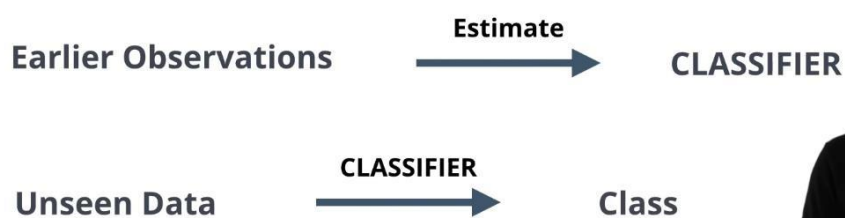## 2. Regression, Classification and Clustering:

Common ML Problems

- **Classification**
- **Regression**
- **Clustering**

Classification Problem:

Classification Problem

Goal: predict category of new observation

Earlier Observations   **Estimate** →   CLASSIFIER

Unseen Data   **CLASSIFIER** →   Class

Classification Applications:

Classification Applications

- **Medical Diagnosis**   Sick and Not Sick
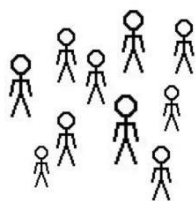
- **Animal Recognition**   Dog, Cat and Horse

Important:

- Qualitative Output

- Predefined Classes

Regression:

Regression

PREDICTORS   ➡   **REGRESSION FUNCTION**   ➡   RESPONSE

- Relationship: **Height** - **Weight**?

- Linear?

- Predict: **Weight** ⟶ **Height**

Regression Model:

Regression Model

Fitting a **linear** function

$Height \approx \beta_0 + \beta_1 \times Weight$

- **Predictor:** Weight
- **Response:** Height
- **Coefficients:** β_0, β_1

Estimate on previous input-output

```
> lm(response ~ predictor)
```

Regression Applications:

Regression Applications

- Payments $\longrightarrow$ Credit Scores
- Time $\longrightarrow$ Subscriptions
- Grades $\longrightarrow$ Landing a Job

- Quantitative Output
- Previous **input-output** observations

Clustering:

Clustering

- **Clustering:** grouping objects in clusters
    - *Similar* within cluster
    - *Dissimilar* between clusters
- **Example:** Grouping similar animal photos
    - No labels
    - No **right** or **wrong**
    - Plenty possible clusterings

k-Means clustering:

k-Means

Cluster data in **k** clusters!
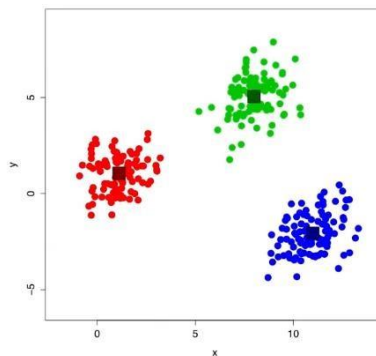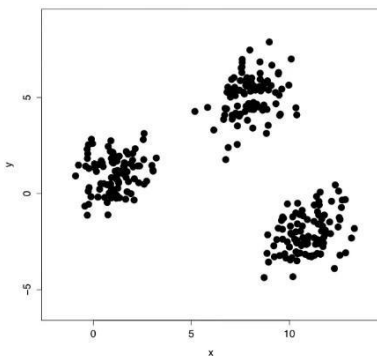
3. **Random Forest Regressor:**

Applications of Random Forest:

Application of Random Forest

Remote Sensing
Used in ETM devices to acquire images of the earth's surface.
Accuracy is higher and training time is less

Object Detection
Multiclass object detection is done using Random Forest algorithms
Provides better detection in complicated environments

Kinect
Random Forest is used in a game console called Kinect
Tracks body movements and recreates it in the game

Random Forest definition:

What is Random Forest?

Random forest or Random Decision Forest is a method that operates by constructing multiple Decision Trees during training phase.
The Decision of the majority of the trees is chosen by the random forest as the final decision

Decision Tree 1 → Output 1

Decision Tree 2 → Output 2

Decision Tree 3 → Output 3

Random Forest
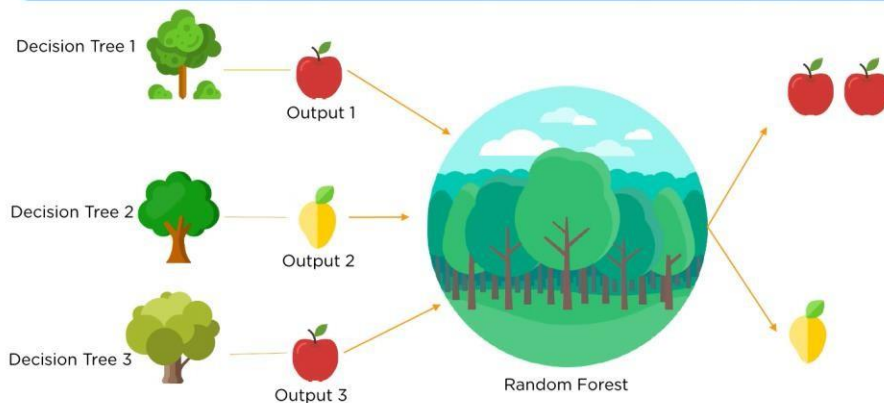
Decision Tree definition:
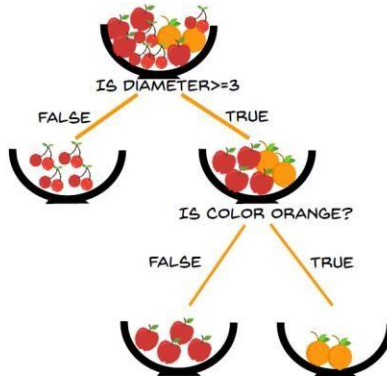
Important terms in decision tree:
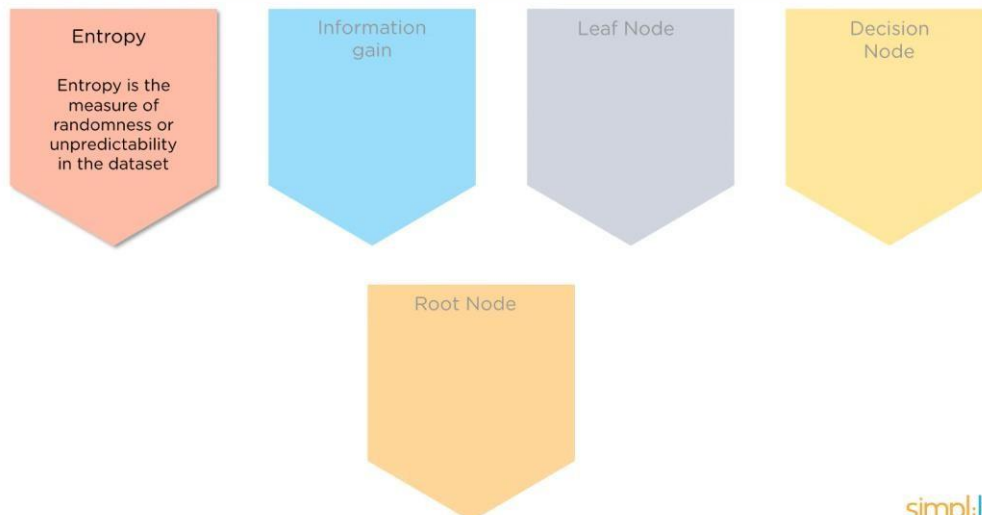
Random Forest Working:
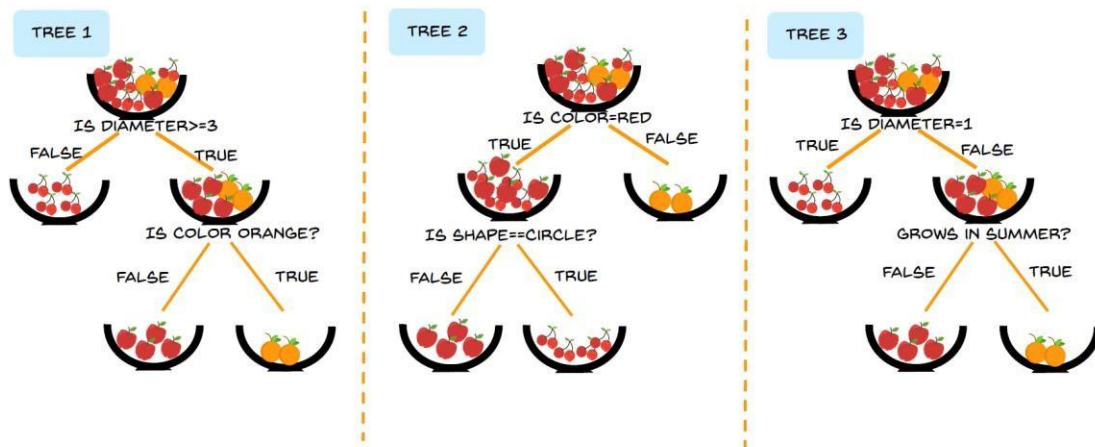
Random Forest using python with iris dataset:

4. Python Flask:

Flask – introduction:

Installation of flask – pre-requisites

Flask Application:

Variable rules in Flask:

Flask – HTTP methods:

Flask uses jinga2 template

Flask – Templates

Flask uses jinga2 template engine

```html
<!doctype html>
<html>
    <body>

        <h1>Hello {{ name }}!</h1>

    </body>
</html>
```

```python
from flask import Flask, render_template
app = Flask(__name__)

@app.route('/hello/<user>')
def hello_name(user):
    return render_template('hello.html', name = user)

if __name__ == '__main__':
    app.run(debug = True)
```

Flask will try to find the HTML file in the templates folder, in the same folder in which this script is present.

The **Jinga2** template engine uses the following delimiters for escaping from HTML

- {% ... %} for Statements
- {{ ... }} for Expressions to print to the template output
- {# ... #} for Comments not included in the template output
- # ... ## for Line Statements