Python pickling.

In python, we sometimes need to ~~xx~~ save the object for later use
this can be done by using python pickle.

Python pickle module is used for "serializing" and "de serializing" a python object structure.
⇒ Any object in python can be pickled so that it can be saved on disk

What pickle does is it serializes the object first before writing it to a file
Pickling is a way to convert a python object in to a "character stream."
The idea is that this character stream contains all the information necessary to reconstruct the object in another python script,

Disadvantages of using pickle in python

(i) Python version Dependency : Data of pickle is so sensitive to the version of python that produced.

(ii) Non - readable : The format of pickle is binary and not easily readable or editable by humans.

pickling with a file of a ML model

* For ML models, "joblib" is often better than pickle

```
import pickle
from sklearn.ensemble import RandomForestClassifier

# Train the sample model
model = RandomForestClassifier()
model.fit (x_train, y_train)

# pickle the model
with open ('model.pkl', 'wb') as file:
    pickle.dump (model, file)
```

with open ('model.pkl', 'wb') as file: ⟶ It opens or creates (if it don't exist) model.pkl in write binary (wb) mode as file

pickle.dump (model, file) ⟶ It dumps the model object into the file

```
# To load the model back
with open ('model.pkl', 'rb') as file :
    loaded_model = pickle.load (file)
```

with open ('model.pkl', 'rb') as file : ⟶ It opens model.pkl in reading binary mode as file

loaded_model = pickle.load (file) ⟶ loads our model in file as loaded_model.