

pass Statement

The `pass` statement is a `null` statement.

- It is generally used as a placeholder for future code i.e. in cases where you want to implement some part of your code in the future but you cannot leave the space blank as it will give a compilation error.
- Sometimes, the `pass` is used when the user does not want any code to execute.
- Using the `pass` statement in loops, functions, class definitions, and `if` statements, is very useful as empty code blocks are not allowed in these.

The syntax of a `pass` statement is:

```
pass
```

Given below is a basic implementation of a conditional using a `pass` statement:

```
n=2
if n==2:
    pass #Pass statement: Nothing will happen
else:
    print ("Executed")
```

In the above code, the `if` statement condition is satisfied because `n==2` is `True`. Thus there will be no output for the above code because once it enters the `if` statement block, there is a `pass` statement. Also, no compilation error will be produced.

Consider another example:

```
n=1
if n==2:
    print ("Executed")
else:
    pass #Pass statement: Nothing will happen
```

In the above code, the `if` statement condition is not satisfied because `n==2` is `False`. Thus there will be no output for the above code because it enters the `else` statement block and encounters the `pass` statement.