

Reloading a module

Let us assume the test.py module contains the following function defined in it.

```
def SayHello(name):  
    print("Hi {}! How are you?".format(name))  
    return
```

We call SayHello() function after importing the module, test.

```
>>> import test  
>>> test.SayHello("Deepak")  
Hi Deepak! How are you?  
>>>
```

Now assume we need to modify the function SayHello() before executing it again. The updated function is:

```
def SayHello(name,course):  
    print("Hi {}! How are you?".format(name))  
    print("Welcome to {} online training program by Internshala".format(course))  
    return
```

In order to call the SayHello() function without ending the current interpreter session, we can reload it by using reload() function from imp module

```
>>> import test  
>>> test.SayHello("Deepak")  
Hi Deepak! How are you?  
>>> import imp  
>>> imp.reload(test)<module 'test' from 'C:\\Python\\test.py'  
>>>> test.SayHello("Deepak", "Python")  
Hi Deepak! How are you?  
Welcome to Python online training program by Internshala  
>>>
```