In [27]:

```
"part one"
class A:
    "Part two"
    def __init__(self,a,b,c):
        "Part three"
        self.assign(a,b,c)
    def assign(self,a,b,c):
        "Part four"
        self.a = a
        self.b = b
        self.c = c
    def pt(self):
        "part five"
        print(self.a, self.b, self.c)

def main():
    q = A(1,2,3)
    print(__doc__)
    print(q.__doc__)
    print(q.__init__.__doc__)
    print(q.assign.__doc__)
    print(q.pt.__doc__)
    q.pt()
if __name__ == "__main__":
    main()
```

```
part one
Part two
Part three
Part four
part five
1 2 3
```

In [28]:

```
""" Hi Hello I am Debanik Roy """
class A:
    """ Welcome To the robotics """
    pass

q = A()
print(__doc__)
print(q.__doc__)
```

```
 Hi Hello I am Debanik Roy
 Welcome To the robotics
```

In [29]:

```
import sys
def main(Q):
    for q in Q:
        print(q)
    return 0
if __name__ == "__main__":
    sys.exit(main(sys.argv))
```

```
C:\Users\Debanik Roy\Anaconda3\lib\site-packages\ipykernel_launcher.py
-f
C:\Users\Debanik Roy\AppData\Roaming\jupyter\runtime\kernel-948e4e1e-5549-40
08-beb3-a009a68b0d54.json

An exception has occurred, use %tb to see the full traceback.

SystemExit: 0



C:\Users\Debanik Roy\Anaconda3\lib\site-packages\IPython\core\interactiveshe
ll.py:3334: UserWarning: To exit: use 'exit', 'quit', or Ctrl-D.
  warn("To exit: use 'exit', 'quit', or Ctrl-D.", stacklevel=1)
```

In [30]:

```
print(__name__)
```

```
__main__
```

In [31]:

```
#operator overloading
class math:
    def __init__(self,a,b,c):
        self.a = a
        self.b = b
        self.c = c
    def __str__(self):
        return ("({0},{1},{2})".format(self.a,self.b,self.c))
    def __add__(self,other):
        self.a += other.a
        self.b += other.b
        self.c += other.c
        return math(self.a,self.b,self.c)
q = math(1,2,3)
q1 = math(2,3,4)
q2 = math(3,4,5)
print(q)
print(q1)
print(q2)
print(q+q1+q2)
```

```
(1,2,3)
(2,3,4)
(3,4,5)
(6,9,12)
```

In [32]:

```
#Method overloading
class A:
    def fun(self, i= None):
        if i is None:
            print("One has exist")
        else:
            print("Two has exist")
q = A()
q.fun()
q.fun(4)
```

```
One has exist
Two has exist
```

In [ ]:

In [ ]: