In [ ]:
```
# defination to handle string input as integers
```

In [2]:
```
def int_validate(user_bal):
    while True:
        if user_bal.isnumeric() == True:
            user_bal = int(user_bal)
            print("balance ",user_bal)
            return user_bal
            break
        else:
            print('Sorry ! Only integers Accepted')
```

In [6]:
```
a = "1111"
b = int_validate(a)
```

```
balance  1111
```

In [7]:
```
b
```

Out[7]:
```
1111
```

In [9]:
```
c = int_validate(a)
```

```
balance  1111
```

In [10]:
```
c
```

Out[10]:
```
1111
```

In [ ]:
```

```

In [11]:
```
# Different ways to handle integers as input
```

isnumeric()

> Note: This method of checking if the string is an integer in Python will not work in negative numbers.

In [13]:
```
# using : isnumeric()

a = "1234"
print(a.isnumeric())
```

```
True
```

In [15]:
```
# checking whether the string is an integer using error handling

flag = True
a = input("enter the number : ")

try:
    int(a)
except ValueError:
    flag = False

if flag:
    print("the string is an integer")
else:
    print("the string is not an integer")
```

```
enter the number : gfdgdf
the string is not an integer
```

In [ ]: