In [10]:

```python
# Example 01
class Rocket:
    def __init__(self, name, distance):
        self.name = name
        self.distance = distance
    def launch(self):
        return "%s has reached %s" % (self.name, self.distance)

class MarsRover(Rocket):
    def __init__(self, name, distance, maker):
        Rocket.__init__(self, name, distance)
        self.maker = maker
    def get_maker(self):
        return "%s Launched by %s" % (self.name, self.maker)

if __name__== "__main__":
    x = Rocket("simple rocket", "till stratosphere")
    y = MarsRover("mars_rover", "till Mars", "ISRO")
    print(x.launch())
    print(y.launch())
    print(y.get_maker())
```

executed in 24ms, finished 16:44:45 2020-07-20

```
simple rocket has reached till stratosphere
mars_rover has reached till Mars
mars_rover Launched by ISRO
```

In [11]:

```python
# Example 02
class MarsRoverComp():
    def __init__(self, name, distance, maker):
        self.rocket = Rocket(name, distance)
        self.maker = maker
    def get_maker(self):
        return "%s Launched by %s" % (self.rocket.name, self.maker)
if __name__== "__main__":
    z = MarsRover("mars_rover2", "till Mars", "ISRO")
    print(z.launch())
    print(z.get_maker())
```

executed in 16ms, finished 16:45:53 2020-07-20

```
mars_rover2 has reached till Mars
mars_rover2 Launched by ISRO
```

In [3]:

```python
# Example 03
class Salary:
    def __init__(self, pay):
        self.pay = pay
    def get_total(self):
        return (self.pay*12)

class Employee:
    def __init__(self, pay, bonus):
        self.pay = pay
        self.bonus = bonus
        self.obj_salary = Salary(self.pay)
    def annual_salary(self):
        return "Total: " + str(self.obj_salary.get_total() + self.bonus)

obj_emp = Employee(200, 800)
print(obj_emp.annual_salary())
```

executed in 24ms, finished 16:31:47 2020-07-20

```
Total: 3200
```

In [13]:

```python
# Task 01
class Birthday:
    def __init__(self, day, month, year):
        self.day = day
        self.month = month
        self.year = year
    def getBirthday(self):
        return f'{self.day}-{self.month}-{self.year}'

class People:
    def __init__(self, name, id, day, month, year):
        self.name = name
        self.id = id
        self.birthday = Birthday(day, month, year)
    def printDetails(self):
        return f'Name: {self.name}\nID: {self.id}\nBirthday: {self.birth
p1 = People('Hamza', 80, '20', 'Dec', '2001')
print(p1.printDetails())
```

executed in 24ms, finished 16:46:57 2020-07-20

```
Name: Hamza
ID: 80
Birthday: 20-Dec-2001
```