



SOLUCIONES

M **MUSK**

EJERCICIO 1

```
class Jet:

    def __init__(self, name, country):
        self.name = name
        self.origin = country

first_item = Jet("F16", "USA")
print(first_item.name)
```

F16

EJERCICIO 2

```
class Jet:

    def __init__(self, name, country):
        self.name = name
        self.origin = country

first_item = Jet("F14", "USA")
second_item = Jet("SU33", "Russia")
third_item = Jet("AJS37", "Sweden")
fourth_item = Jet("Mirage2000", "France")
fifth_item = Jet("Mig29", "USSR")
sixth_item = Jet("A10", "USA")
```

EJERCICIO 3

```
class Jet:

    def __init__(self, name, country, quantity):
        self.name = name
        self.origin = country
        self.quantity

first_item = Jet("F14", "USA", 87)
second_item = Jet("Mirage2000", "France", 35)
```

EJERCICIO 4

```
class Nobel:

    def __init__(self, category, year, winner):
        self.category = category
        self.year = year
        self.winner = winner
```

EJERCICIO 5

```
class Student:  
    def __init__(self, name, age, grade):  
        self.name = name  
        self.age = age  
        self.grade = grade  
  
e = Student('Alan', 16, '11th')
```

EJERCICIO 6

```
class Student:  
    pass  
  
e = Student()
```

EJERCICIO 7

```
from statistics import mean

class Student:
    def __init__(self, name, age, grade):
        self.name = name
        self.age = age
        self.grade = grade

    def compute_avg_grade(self, marks):
        self.grade = mean(marks)

e = Student('Alan', 16, '11th')
e.compute_avg_grade([6, 7, 6.5, 8])
print(e.grade)
```

6.875

EJERCICIO 8

```
from statistics import mean

class Student:
    def __init__(self, name, age, grade):
        self.name = name
        self.age = age
        self.grade = grade

    @staticmethod
    def get_non_approved_subjects(marks):
        for subject, mark in marks.items():
            if mark < 5:
                print('Subject of {} has been failed.'.format(subject))

    def compute_avg_grade(self, marks):
        self.grade = mean(marks)

Student.get_non_approved_subjects({'maths': 6, 'history': 4.5, 'biology': 7, 'physics': 4})
```

Subject of history has been failed.

Subject of physics has been failed.

EJERCICIO 9

```
from statistics import mean

class Student:
    school = 'UB'

    def __init__(self, name, age, grade):
        self.name = name
        self.age = age
        self.grade = grade

    @staticmethod
    def get_non_approved_subjects(marks):
        for subject, mark in marks.items():
            if mark < 5:
                print('Subject of {} has been failed.'.format(subject))

    @classmethod
    def change_school(cls, school):
        cls.school = school

    def compute_avg_grade(self, marks):
        self.grade = mean(marks)

e = Student('Alan', 16, '11th')
print(e.school)
Student.change_school('UPC')
print(e.school)
```

UB
UPC

EJERCICIO 10

```
class Student:
    school = 'UB'

    def __init__(self, name, age, grade):
        self.name = name
        self.age = age
        self.grade = grade

    def _compute_constant_assistance_mark(self, assistance):
        mark = 0
        assistance_list = list(assistance.values())

        for a in assistance_list:
            if a < 4:
                mark = 1
            return mark
            elif a >= 4 and a < 8:
                mark = 2

        if mark == 0:
            mark = 3
        return mark

    def test_function(self, assistance):
        return self._compute_constant_assistance_mark(assistance)

e = Student('Alan', 16, '11th')
print(e.test_function({'January': 5, 'February': 2, 'March': 0, 'April': 10}))
print(e.test_function({'January': 6, 'February': 7, 'March': 5, 'April': 7}))
print(e.test_function({'January': 8, 'February': 12, 'March': 10, 'April': 15}))
```

1
2
3



MUSK

The logo consists of a stylized letter 'M' on the left, composed of three vertical bars with small horizontal lines at the top and bottom. To its right, the word 'MUSK' is written in a bold, white, sans-serif font.