1. Write a function in Python with a string such that it accepts a parameter- “stringsplit”. This encoded string will contain your name, domain name and register number. You can separate the values in the string by any number of underscores. [The string should not contain any other underscore symbols in your name, domain name and register number]. The function should return a Python dictionary with your name, domain name and register number. For example, if the input would be " Aaron__Googleplaystore___2347201”. Then the function should return the output as follows: { "name": " Aaron ", "Domain_name": " Googleplaystore ", "Regno": "2347201" }

In [ ]:

```python
def decode(stringsplit):
    str = stringsplit.split('_')
    str = [str.strip() for str in str if str.strip()]
    dict = {
        "name": str[0],
        "Domain_name": str[1],
        "Regno": str[2]
    }
    return dict
encode= "Shubham___DocumentManagementSystem____2347256"
dict = decode(encode)
print(dict)
```

```
{'name': 'Mishra', 'Domain_name': 'DocumentManagementSystem', 'Regno': '2347256'}
```

2. Write a Python program to implement the object-oriented concepts of multiple, Multilevel and Hierarchical Inheritances using your domain applications.

In [ ]:

```python
# defining User Parent Class
class User:
    def __init__(self, username, role):
        self.username = username
        self.role = role
    def display_info(self):
        print(f"Username: {self.username}, Role: {self.role}")
#Defining Different Roles for the User
class Admin(User):
    def __init__(self, username):
        super().__init__(username, "Admin")

    def manage_documents(self):
        print(f"{self.username} is managing documents.")


class RegularUser(User):
    def __init__(self, username):
        super().__init__(username, "Regular User")

    def view_documents(self):
        print(f"{self.username} is viewing documents.")


class Editor(RegularUser):
    def __init__(self, username):
        super().__init__(username)
        self.role = "Editor"

    def edit_documents(self):
        print(f"{self.username} is editing documents.")


class Viewer(RegularUser):
    def __init__(self, username):
        super().__init__(username)
        self.role = "Viewer"

    def view_only(self):
        print(f"{self.username} can only view documents.")


# Createing instances and demonstrateing the relationships
admin = Admin("admin_user")
editor = Editor("editor_user")
viewer = Viewer("viewer_user")

print("Admin")
admin.display_info()
admin.manage_documents()

print("Editor")
editor.display_info()
editor.view_documents()
editor.edit_documents()

print("Viewer")
viewer.display_info()
viewer.view_documents()
viewer.view_only()
```

```
Admin
Username: admin_user, Role: Admin
admin_user is managing documents.
Editor
Username: editor_user, Role: Editor
editor_user is viewing documents.
editor_user is editing documents.
Viewer
Username: viewer_user, Role: Viewer
viewer_user is viewing documents.
viewer_user can only view documents.
```