Section: A **Quiz 22** **Roll number:** BSSB23001

Q1 You have been provided with Developer and GameGenre classes to aid in a video game management system. Developers and genres can exist independently of any specific video game. You need to establish an aggregation relationship between the VideoGame class and the Developer and GameGenre classes.

```
class Developer {
private:
string name;
string location;
public:
Developer(string name, string location) {
this->name = name;
this->location = location;
}
string getDevName();
string getLocation();
};

class GameGenre {
private:
string name;
string description;
public:
GameGenre(string name, string description) {
this->name = name;
this->description = description;
}
string getGenreName();
string getDescription();
};
```

10

1. ***You need to code for the VideoGame class, which will demonstrate an aggregation relationship between the VideoGame class and the Developer and GameGenre classes. The VideoGame class should have title and playtime (in hours) as private attributes. You will need to write a parameterized constructor, destructor, and a display function to showcase the details of a video game, including its title, developer's name, location, genre, description, and playtime.*** ***(6)***
2. ***Demonstrate its use in the main function and display the video game details using the display function.*** ***(4)***

```
#include <iostream>

using namespace std;

// considering above other classes

class VideoGame {
    string title;
    int playtime;
    Developer * developer;
    GameGenre * gameGenre;
```

```
public:
    VideoGame(const string& _tit, const int& _pt, Developer* devel,
                                                GameGenre* gameGen) {
        title = _tit;
        playtime = _pt;
        developer = devel;
        gameGenre = gameGen;
    }
    ~VideoGame = default;
    void display() const {
        cout << "Title: " << title << endl;
        cout << "Playtime: " << playtime << endl;
        cout << "DeveloperName: " << developer->getDevName() << endl;
        cout << "Developer Location: " << developer->getLocation() << endl;
        cout << "Genre: " << gameGenre->getGenreName() << endl;
        cout << "Description: " << gameGenre->getDescription() << endl;
    }
};

int main() {
    Developer* developer("HD", "PC");
    GameGenre gameGenre("PH", "This is Genre");
    VideoGame game("GTA", 20, &developer, &gameGenre);
    game.display();
    cout << "This is sample implementation\n";

    return 0;
};
```