

# C++: STROIKA GAME

```
// Es crea el constructor de BasePlayer().
BasePlayer::BasePlayer() {
    ...
}

// Es crea un constructor que servirà per inicialitzar un objecte d'una determinada classe al mateix temps en què es declara.
BasePlayer::BasePlayer(float custom_life) {
    life = custom_life;
}

// Es crea un destructor que s'invocarà automàticament quan l'objecte surti de l'abast o sigui destruït explícitament.
BasePlayer::~BasePlayer() {
}

// Mètode per "rebre damage" que té com a variable d'entrada un float pel damage.
void BasePlayer::RecieveDamage(float damage) {
    life -= damage;
}

// Mètode que usa un punter i que és usat per aplicar el damage causat.
void BasePlayer::ApplyDamage(BasePlayer* punter, float damage) {
    punter->RecieveDamage(damage);
}

// Getter per poder agafar la vida.
float BasePlayer::getLife() {
    return life;
}
```

*BasePlayer.cpp*

C++ on es crea el constructor del BasePlayer() i on s'inicialitza.

Es crea el constructor i el destructor del BasePlayer() i es creen dues funcions, una per rebre la quantitat de vida que se li restarà al BasePlayer() i una altre funció que servirà per aplicar la quantitat de vida causada.

També disposa d'un getter() per poder agafar la vida del BasePlayer().

```
class BasePlayer
{
    // Atribut protected de "life".
protected:
    float life;

public:

    // S'instancien els BasePlayer() i els seus atributs.
    BasePlayer();
    BasePlayer(float custom_life);
    ~BasePlayer();

    // S'instancien les funcions creades al BasePlayer i el getter de life.
    void RecieveDamage(float damage);
    void ApplyDamage(BasePlayer* punter, float damage);
    float getLife();
};
```

*BasePlayer.h*

Arxiu de capçalera (classe) que crea l'atribut protected de vida i on s'instancien els BasePlayer() amb els seus atributs. També és on s'instancien les funcions creades al BasePlayer() i a l'anterior getter() de la vida.

```

int main()
{
    // Se li aplica 300 de vida al Player.
    BasePlayer player(300.0f);

    // Se li aplica 200 de vida a l'enemic.
    BasePlayer EnrageEnemy(200.0f);

    BasePlayer* punterPlayer = &player;

    // El Player rebrà la quantitat de 40 d'atac, que és el que té EnrageEnemy.
    EnrageEnemy.ApplyDamage(punterPlayer, 40.0f);

    // Es mostra per consola la vida que té el Player.
    printf("PRIMERS EXERCICIS: ");
    printf("\n");

    cout << "PLAYER LIFE: " << punterPlayer->getLife() << endl;
}

```

*AppMain.cpp*

Se li apliquen primerament 300.0f de vida al Player i 200.0f de vida a l'EnrageEnemy(). S'executa el mètode ApplyDamage() i se li resta 40.0f de vida a l'enemic. Finalment es mostra la vida de l'enemic.

```

printf("LAST EXERCISE: ");
printf("\n");
// Variable on es desarà el text de l'usuari.
char stringText[50], charValue;
int comptadorX;

// Es formula la primera pregunta. En aquest cas no és possible dir que no.
printf("Enter your username to start the game: ");
comptadorX = 0;
charValue = getchar();

printf("");
printf("");

// Bucle que espera a que li facis ENTER i que captura el que escriu l'usuari
while (charValue != '\n')
{
    stringText[comptadorX] = charValue;
    comptadorX++;
    charValue = getchar();
}

stringText[comptadorX] = '\0';

```

```

// Mostra el text del nom d'usuari i pregunta si l'usuari està segur que vol jugar.
printf("You are logged in as the user: %s", stringText);
printf(". Are you sure you want to start the game?");
printf("");

// Pregunta a l'usuari si vol jugar o no.
printf("\nYES or NO: \n");
char resultatJugar;
resultatJugar = getchar();

// Si el resultat és "YES" o semblant, entra al condicional que explica la història.
if (resultatJugar == 'Y' || resultatJugar == 'y'
    || resultatJugar == 'Yes' || resultatJugar == 'yes' || resultatJugar == 'YES')
{
    // Es crea el BasePlayer que rebrà el dany en aquesta partida.
    BasePlayer EnrageEnemy2(150.0f);
    BasePlayer* punterPlayer2 = &player;
    EnrageEnemy2.ApplyDamage(punterPlayer2, 200.0f);
}

```

```
printf("\nSo the user: %s", stringText, "%s has grabbed a party cubata and threw it over his head.");
printf("\nArnau is not dead, but he will be in the hospital for a long time and will not be able to continue with his misdeeds.");
printf("\nArnau's life in the hospital is: %f", punterPlayer->getLife());
printf(" of life.");

printf("\n ");

printf("\nThank you so much for being by our side in this fight. We will never forget you. VISCA CATALUNYA!");

printf("\n ");
printf("\n ");
```

Es tracta d'un joc creat per consola, el qual inicialment et demana el teu nom d'usuari, després et fa una pregunta sobre si realment vols jugar a la partida i, si escrius que sí, et mostra una història aplicant totes les mecàniques necessàries per dur a terme aquesta pràctica.