

```

✓ #ifndef CALCULATOR_H
#define CALCULATOR_H

✓ #include<iostream>
#include <string>

using namespace std;

✓ class Player {
private:
:   string player[2];
public:
:   void setPlayer();
:   string *getPlayer();
};

✓ class GamblingGame {
private:
:   int ranNum[3];
public:
:   void gameStart(string p[2]);
};

#endif

```

일단 제시된 클래스2개를 먼저 만들어 주었다.

Player클래스에는 플레이어의 이름을 배열로 받아야하니 string으로 player라는 배열을 만들었고.

player의 이름을 받을 setPlayer(), 배열 player를 반환할 getPlayer()도 선언하였다. 여기서 getPlayer()에 포인터를 사용한 이유는 배열에 포인터를 사용하여 배열의 첫번째 요소의 주소를 반환해 밖에서 참조할 수 있게 하였다.

그다음 GamblingGame클래스를 만든다음

랜덤으로 나온 숫자 3개를 받을 배열 ranNum[3]을 선언하였고 gameStart()에는 string p[2]라는 요소를 받게 하였는데 이는 getPlayer로 얻은 배열을 사용하기 위해 사용하였다.

```

#include <iostream>
#include <string>
#include <random>
#include <ctime>

#include "game.h"

using namespace std;

void Player::setPlayer() {
    cout << "첫번째 선수 이름>>";
    getline(cin, player[0], '\n');
    cout << "두번째 선수 이름>>";
    getline(cin, player[1], '\n');
}

string Player::getPlayer() {
    return player;
}

void GameEngine::gameStart(string n[2]) {
    srand(time(NULL));
    string enter[1];
    do {
        for (int i = 0; i < 2; i++) {
            cout << n[i] << " Enter>";
            getline(cin, enter[i]);
        }
        if (enter[0] == " ") {
            for (int i = 0; i < 3; i++) {
                int random = rand() % 3;
                ranNum[i] = random;
                cout << random << " ";
            }
            if (ranNum[0] == ranNum[1] && ranNum[1] == ranNum[2]) {
                cout << n[1] << " 님 승리!" << endl;
                return;
            }
            else {
                cout << "아쉽군요!" << endl;
            }
        }
    } while (true);
}

```

해당 클래스의 구현부에는 setPlayer()에 getline()을 사용하여 배열에 선수 두명의 이름을 받은다음 getPlayer로 배열player를 반환할 수 있게 하였다.

gameStart에는 게임에서 진행되는 과정을 담았는데 랜덤한 수 3개를 받아야하기에 rand라이브러리를 사용하였다. srand(time(NULL))은 해당 rand함수를 실행할 때 해당시간의 값을 seed값으로 갖게하고 시간은 항상 변하기에 seed값도 변하여 랜덤한 값을 뽑을 수 있게 해준다. 만약 하지 않는다면 seed값이 같아 rand함수는 동일한 값만을 출력할 것이다. 그다음 enter라는 문자열을 선언해 주었는데. 이는 엔터를 눌렀을 때 실행하기 위해 엔터를 눌렀다는 것을 확인할 문자열이다. do-while문을 사용하여 승리자가 나올 때 까지 무한반복하게 하였고, 그다음 for문으로 총 2명이 뽑아야하니 2번 반복하게 한다음 p[j]로 선수를 구분하였는데 string p[2]에 배열player를 받을 것이기에 p[0]은 첫번째. p[1]은 두번째로 구분이 된다. 그다음 getline을 사용 엔터를 누르면 문자열enter에 공백인 문자열이 생기기에 if문으로 enter[0]에 공백이 있는지 확인하고 공백이 있으면 엔터를 누른것으로 간주한다 for문으로 랜덤한 0부터2까지의 숫자를 뽑은다음 배열 ranNum에 저장하고 해당 숫자를 출력한다. 그러면 ranNum은 숫자 3개를 가진 배열이 되는데 이 숫자 3개를 if문을 사용해 비교하여 3개가 다 같으면 p[j](번째 플레이어)가 승리했다는 문구를 출력후 return하여 함수를 종료한다. 만약 다르면 "아쉽군요!"라고 출력한다음 다시 루프를 돈다.

```

✓ #include <iostream>
  #include "gamble.h"

✓ int main()
{
    Player player;
    player.setPlayer();

    GamblingGame game;
    game.gameStart(player.getPlayer());
}

```

Player클래스의 객체 player를 만들고  
 setPlayer()로 플레이어의 이름을 받는다음  
 GamblingGame의 객체 game을 만들고  
 game.gameStart를 하는데 플레이어의 이름이  
 필요하니 player.getPlayer()로 플레이어들의 이름이  
 들어간 배열을 준다.

```

첫번째 선수 이름>>신윤섭
두번째 선수 이름>>홍길동
신윤섭: <Enter>
0 2 1 아침균요!
홍길동: <Enter>
2 0 0 아침균요!
신윤섭: <Enter>
1 0 0 아침균요!
홍길동: <Enter>
2 1 2 아침균요!
신윤섭: <Enter>
1 0 0 아침균요!
홍길동: <Enter>
2 0 0 아침균요!
신윤섭: <Enter>
2 0 0 아침균요!
홍길동: <Enter>
1 2 0 아침균요!
신윤섭: <Enter>
0 0 2 아침균요!
홍길동: <Enter>
0 1 2 아침균요!
신윤섭: <Enter>
0 1 2 아침균요!
홍길동: <Enter>
2 1 0 아침균요!
신윤섭: <Enter>
2 1 1 아침균요!
홍길동: <Enter>
2 1 2 아침균요!
신윤섭: <Enter>
0 2 0 아침균요!
홍길동: <Enter>
1 1 2 아침균요!
신윤섭: <Enter>
0 0 2 아침균요!
홍길동: <Enter>
2 0 1 아침균요!
신윤섭: <Enter>
1 2 1 아침균요!
홍길동: <Enter>
2 1 2 아침균요!
신윤섭: <Enter>
0 2 0 아침균요!
홍길동: <Enter>
0 0 0 홍길동님 승리!

```

# 실행결과

202104231 신윤섭