**Assignment 9**

1. Create a class that captures planets. Each planet has a name, a distance from the sun, and its gravity relative to Earth’s gravity. For distance and gravity, use the type double which captures real numbers. Make objects for Earth and your favorite non-earth planet.

Code:

#include <iostream>

#include <string>

using namespace std;

class Planets{

string p\_name;

double gravity,dsun;

public:

void input(){

cout << "Enter the name of the planet: " << endl;

cin >> p\_name;

cout << "Enter distance from the sun: " << endl;

cin >> dsun;

cout << "Enter gravity relative to the earth: " <<endl;

cin >> gravity;

}

void display(){

cout << "Planet Name: " << p\_name <<endl;

cout << "Distance from the sun: " << dsun<<"kms"<<endl;

cout << "Relative Gravity: " <<gravity<<"m/s\*\*2" <<endl;

}

};

int main()

{

Planets earth,ne;

earth.input();

earth.display();

ne.input();

ne.display();

return 0;

}

Output:

Graphical user interface, text, application

Description automatically generated

1. Write a C++ Program to Display today’s Date using Constructor.

Code:

#include<iostream>

using namespace std;

class date

{

private:

int dd, mm, yy;

public:

date()

{

dd=27;

mm=9;

yy=2022;

cout<<"\nObject Created.\n";

}

void display()

{

cout<<"\nThe Entered Date is :: ";

cout<<dd<<"-"<<mm<<"-"<<yy<<"\n";

}

};

int main ()

{

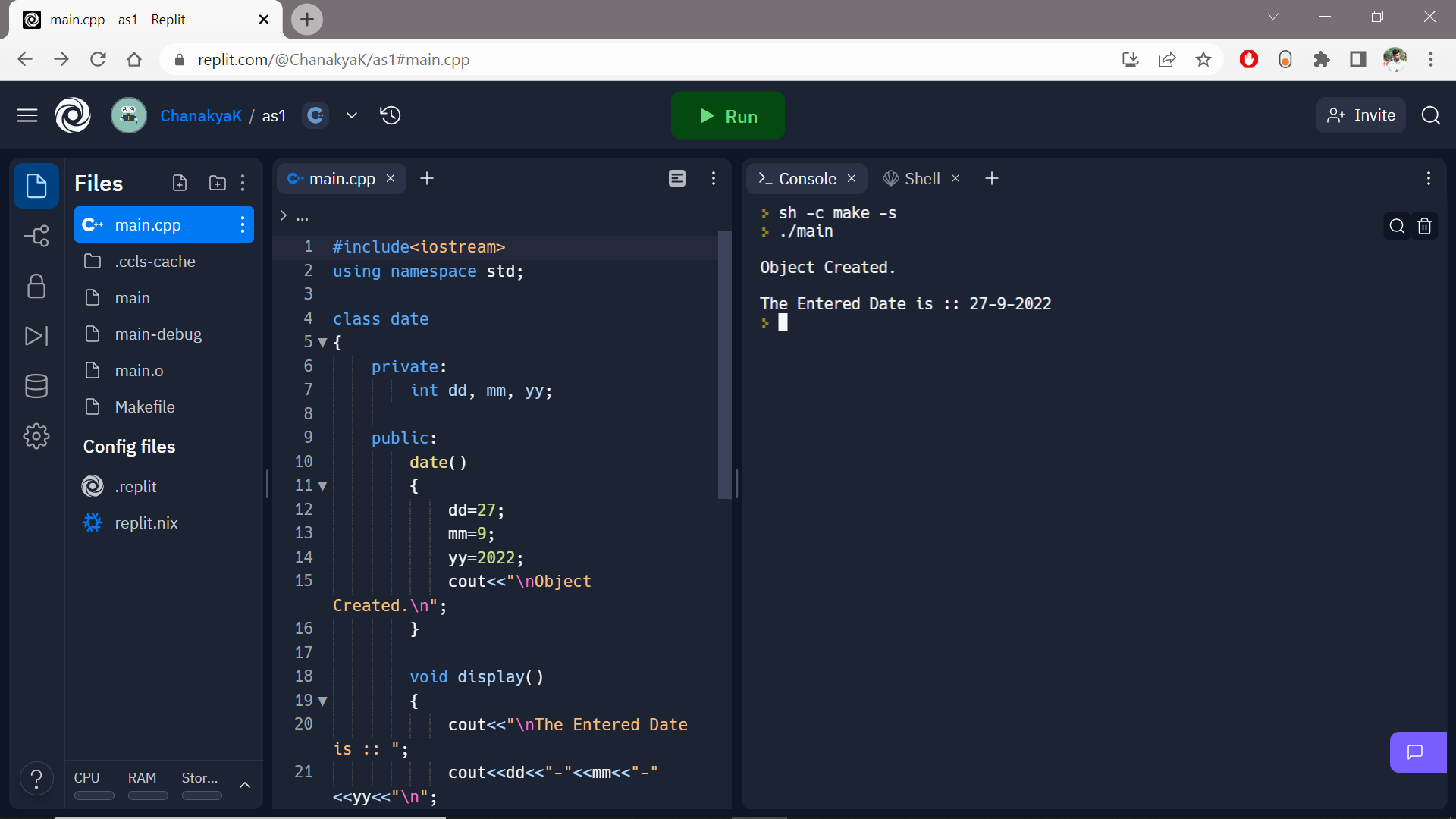
date d1;

d1.display ();

return 0;

}

Output:



1. Write a C++ program with all access specifiers and show its usage in program.

**Code:**

**#include <iostream>**

**using namespace std;**

**class Parent**

**{**

**protected:**

**int id\_key;**

**};**

**class Child : public Parent**

**{**

**public:**

**void setId(int id)**

**{**

**id\_key = id;**

**}**

**void displayId()**

**{**

**cout << "id\_key is: " << id\_key << endl;**

**}**

**};**

**int main() {**

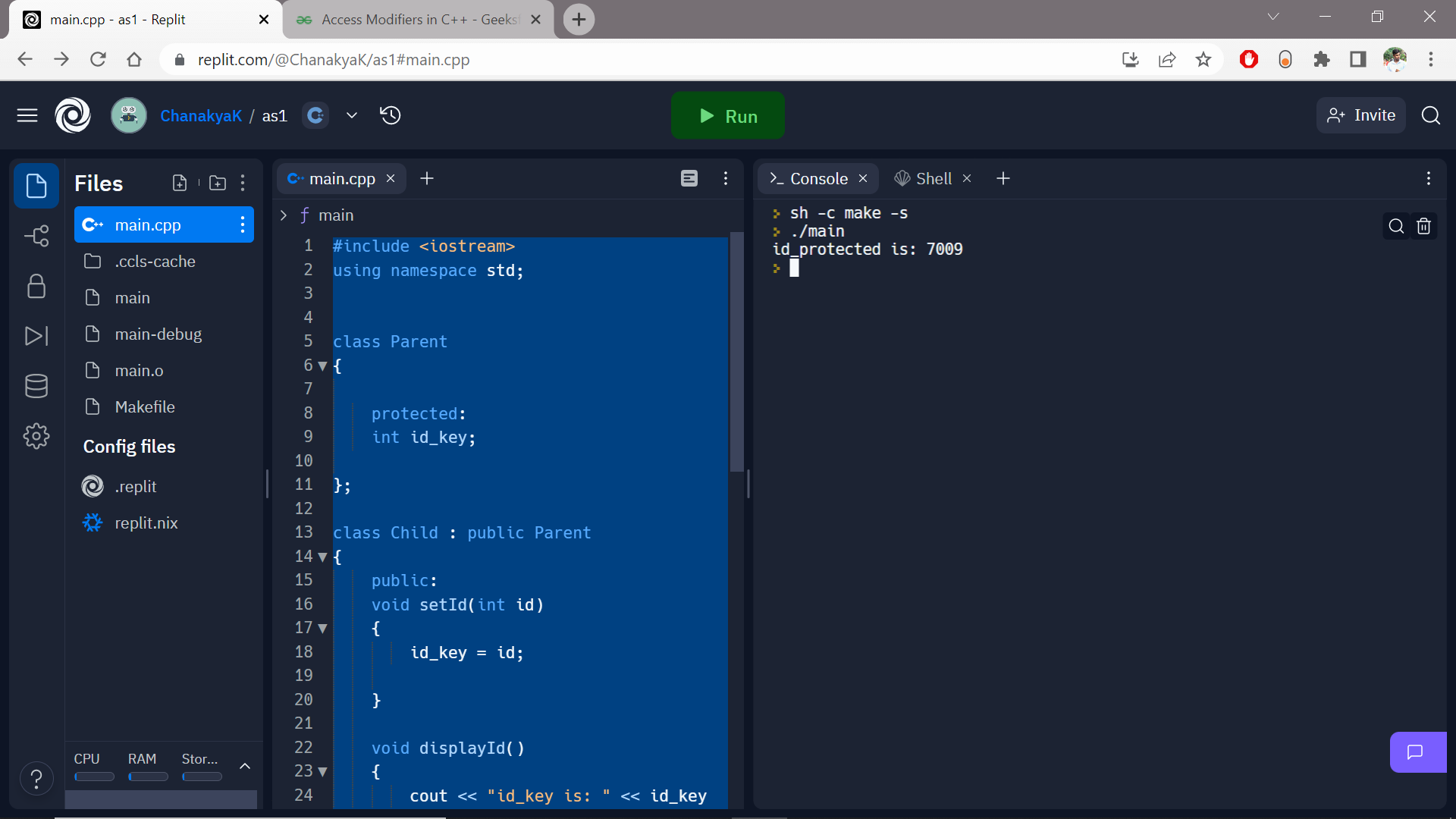
**Child obj1;**

**obj1.setId(7009);**

**obj1.displayId();**

**return 0;**

**}**

**Output:** 

1. Write a program in C++ to find the sum of the input series using friend function.

*Example:*

*Enter the values:*

*1*

*2*

*111*

*Sum is: 114*

**Code:**

**#include<iostream>**

**#include<math.h>**

**using namespace std;**

**class Sum{**

**int summation;**

**friend void soN(Sum &s);**

**public:**

**void display(){**

**cout<<"The total sum is " <<summation<<endl;**

**}**

**};**

**void soN(Sum &s)**

**{**

**int n;**

**cout <<"Enter the positive number " <<endl;**

**cin >> n;**

**s.summation = 0;**

**if(n>0)**

**do**

**{**

**s.summation += n;**

**cout<<"Till now the sum is " << s.summation <<endl;**

**cout<<"Enter another number "<<endl;**

**cin >> n;**

**} while (n>0);**

**cout<<"The entered number is negative \n the required sum till now is "<<endl;**

**}**

**int main(){**

**Sum s;**

**soN(s);**

**s.display();**

**return 0;**

**}**

**Output:**

Text

Description automatically generated

MCQs

Where friend keyword placed for friend function?

(A). Function definition

(B). Function declaration

(C). Main function

(D). Both A and C

Global Function can be Friend functions for which of the following class?

(A). Base class

(B). Derived class

(C). Any classes

(D). None of these