

REPORT ON JAVA INTERFACES

(COMPILER DESIGN)

Definition: An interface is a reference type in Java. It is similar to class. It is a collection of abstract methods.

Abstract method: A method without body (no implementation) is known as abstract method.

Ex : `public int myMethod(int n1, int n2);`

Features:

1. Along with abstract methods, an interface may also contain constants, default methods, static methods, and nested types.
2. Method bodies exist only for default methods and static methods.
3. An interface can contain any number of methods.
4. An interface is written in a file with a ".java" extension, with the name of the interface matching the name of the file.
5. The byte code of an interface appears in a ".class" file.
6. Interfaces appear in packages, and their corresponding bytecode file must be in a directory structure that matches the package name.
7. We cannot instantiate an interface.
8. An interface does not contain any constructors.

Declaring Interfaces:

The interface keyword is used to declare an interface. Here is a simple example to declare an interface –

```
* File name : NameOfInterface.java */  
  
import java.lang.*;  
  
// Any number of import statements  
  
public interface NameOfInterface {  
  
    // Any number of final, static fields
```

```
    // Any number of abstract method declarations\  
}
```

Properties of Interface:

1. An interface is implicitly abstract. We do not need to use the abstract keyword while declaring an interface.
2. Each method in an interface is also implicitly abstract, so the abstract keyword is not needed.
3. Methods in an interface are implicitly public.

Java Interface Implementation

Syntax:

```
public interface NameOfInterface {  
    // Any number of final, static fields  
    // Any number of abstract method declarations\  
}
```

Example

```
// Filename: Sports.java  
public interface Sports {  
    public void setHomeTeam(String name);  
    public void setVisitingTeam(String name);  
}
```

Things to check:

1. opening and closing braces to be balanced
2. syntax appropriately written

Limitations:

1. You can only use char, String, int and float values to declare any function

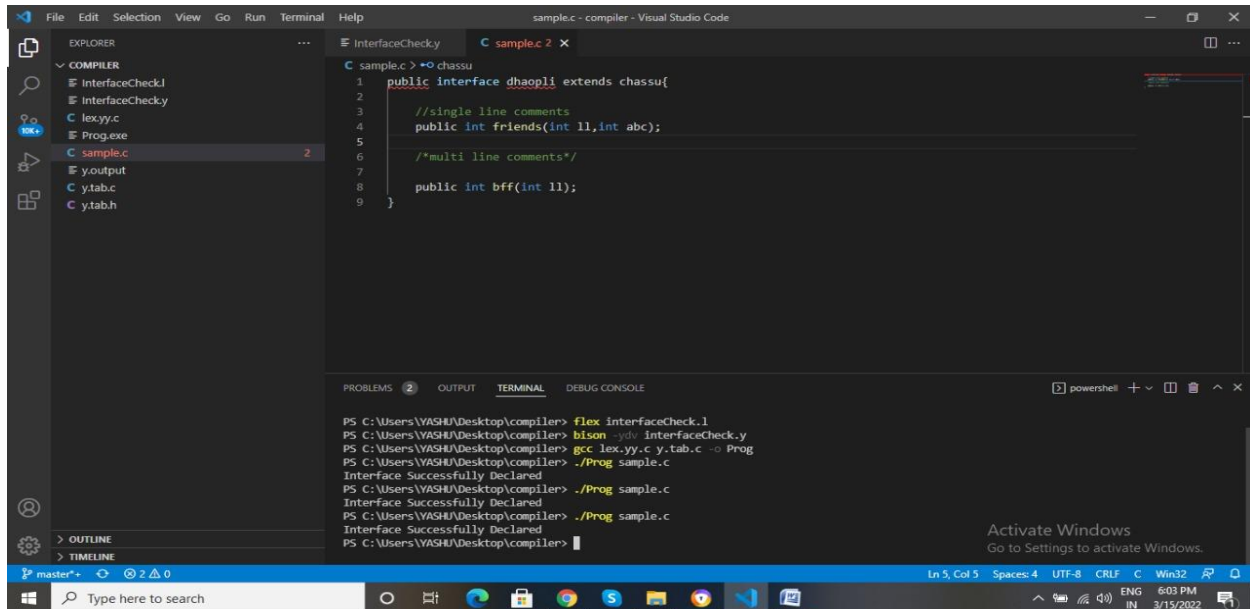
Erros not handled:

1. Anthing after closing bracket of interface is not handled.

Test Cases :

Accepted test cases

1. Handled single line and multiline comments.

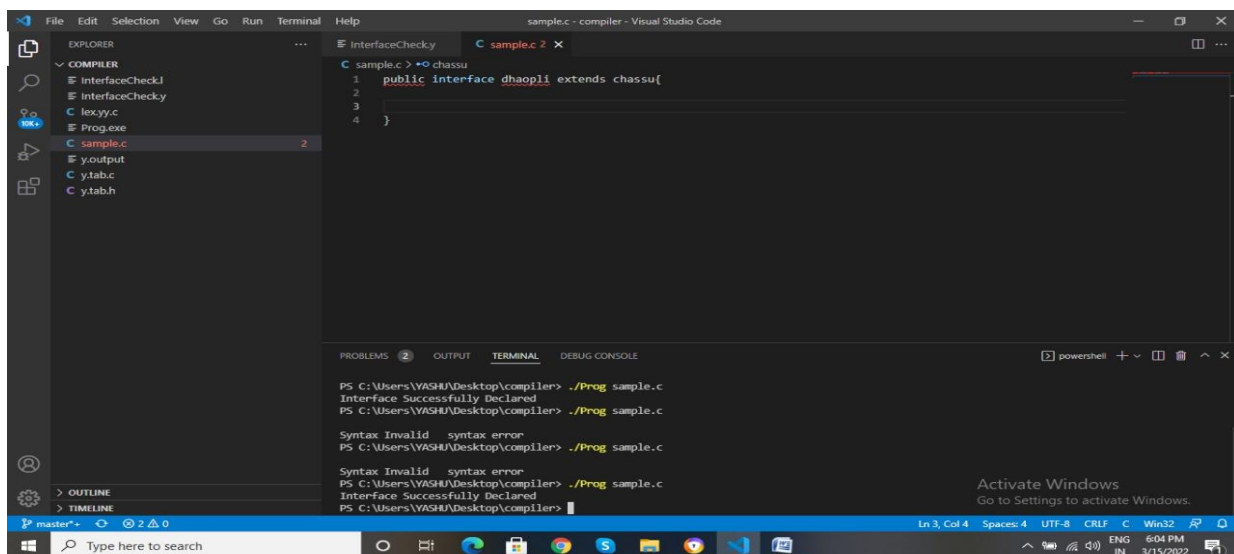


The screenshot shows the Visual Studio Code editor with a C program named `sample.c` open. The program defines a public interface `InterfaceCheck1` that extends `chassu`. It includes a single-line comment `//single line comments` and a multi-line comment `/*multi line comments*/`. The interface contains two functions: `public int friends(int ll, int abc);` and `public int bff(int ll);`. The terminal window shows the compilation process using `flex`, `bison`, and `gcc`, resulting in a successful declaration of the interface.

```
sample.c > *O chassu
1 public interface dhaopli extends chassu{
2
3 //single line comments
4 public int friends(int ll, int abc);
5
6 /*multi line comments*/
7
8 public int bff(int ll);
9 }
```

```
PS C:\Users\YASHU\Desktop\compiler> flex InterfaceCheck.1
PS C:\Users\YASHU\Desktop\compiler> bison -ydv InterfaceCheck.y
PS C:\Users\YASHU\Desktop\compiler> gcc lex.yy.c y.tab.c -o Prog
PS C:\Users\YASHU\Desktop\compiler> ./Prog sample.c
Interface Successfully Declared
PS C:\Users\YASHU\Desktop\compiler> ./Prog sample.c
Interface Successfully Declared
PS C:\Users\YASHU\Desktop\compiler> ./Prog sample.c
Interface Successfully Declared
PS C:\Users\YASHU\Desktop\compiler>
```

2. Handled after removing body of interface .

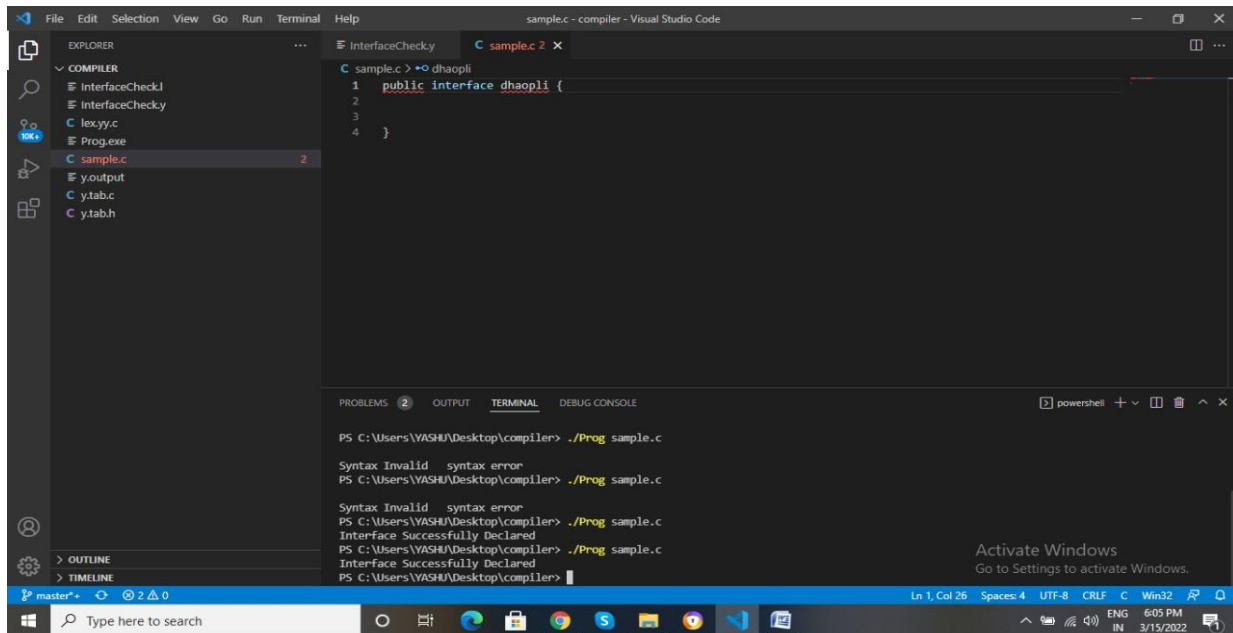


The screenshot shows the Visual Studio Code editor with a C program named `sample.c` open. The program defines a public interface `InterfaceCheck1` that extends `chassu`. The interface contains a single function declaration: `public int friends(int ll, int abc);`. The terminal window shows the compilation process using `flex`, `bison`, and `gcc`, resulting in a successful declaration of the interface.

```
sample.c > *O chassu
1 public interFace dhaopli extends chassu{
2
3
4 }
```

```
PS C:\Users\YASHU\Desktop\compiler> ./Prog sample.c
Interface Successfully Declared
PS C:\Users\YASHU\Desktop\compiler> ./Prog sample.c
Syntax Invalid syntax error
PS C:\Users\YASHU\Desktop\compiler> ./Prog sample.c
Syntax Invalid syntax error
PS C:\Users\YASHU\Desktop\compiler> ./Prog sample.c
Interface Successfully Declared
PS C:\Users\YASHU\Desktop\compiler>
```

3. Handled without extends keyword



The screenshot shows the Visual Studio Code editor with a C file named `sample.c`. The code defines a public interface `dhaopli` with a single function `friends` that takes an integer `a` and returns an integer. The terminal output shows the command `./Prog sample.c` being executed, resulting in the message "Interface Successfully Declared".

```
sample.c > +0 dhaopli
1 public interface dhaopli {
2
3
4 }
```

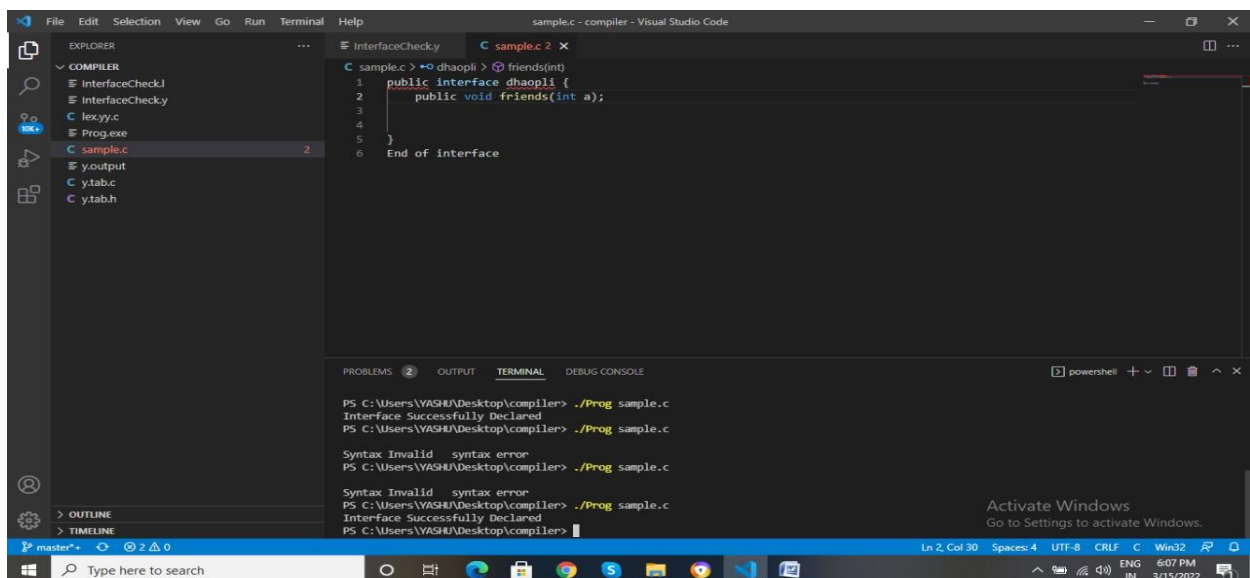
```
PS C:\Users\YASHU\Desktop\compiler> ./Prog sample.c

Syntax Invalid syntax error
PS C:\Users\YASHU\Desktop\compiler> ./Prog sample.c

Syntax Invalid syntax error
PS C:\Users\YASHU\Desktop\compiler> ./Prog sample.c
Interface Successfully Declared
PS C:\Users\YASHU\Desktop\compiler> ./Prog sample.c
Interface Successfully Declared
PS C:\Users\YASHU\Desktop\compiler>
```

4. Not handled – anything after closing bracket of interface is accepted

Assumptions – after declaring interface we can declare class which implements this interface , so its accepted.



The screenshot shows the Visual Studio Code editor with a C file named `sample.c`. The code defines a public interface `dhaopli` with a single function `friends` that takes an integer `a` and returns an integer. Below the interface, a class `friends` is declared, which implements the `friends` function. The terminal output shows the command `./Prog sample.c` being executed, resulting in the message "Interface Successfully Declared".

```
sample.c > +0 dhaopli > +0 friends(int)
1 public interface dhaopli {
2     public void friends(int a);
3
4
5 }
6 End of interface
```

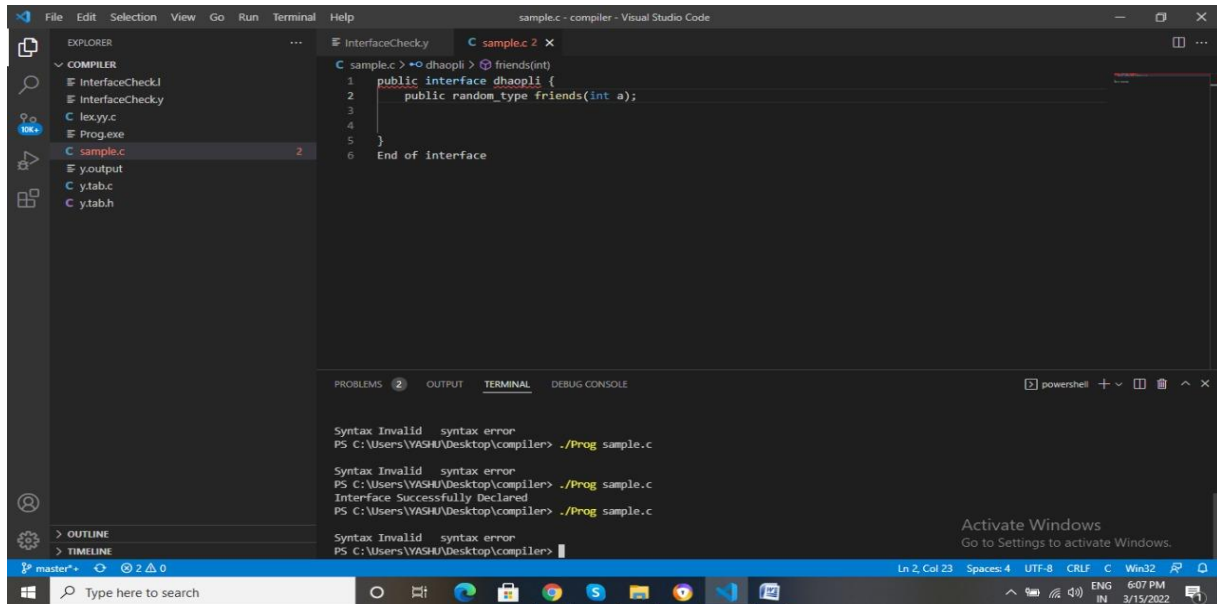
```
PS C:\Users\YASHU\Desktop\compiler> ./Prog sample.c
Interface Successfully Declared
PS C:\Users\YASHU\Desktop\compiler> ./Prog sample.c

Syntax Invalid syntax error
PS C:\Users\YASHU\Desktop\compiler> ./Prog sample.c

Syntax Invalid syntax error
PS C:\Users\YASHU\Desktop\compiler> ./Prog sample.c
Interface Successfully Declared
PS C:\Users\YASHU\Desktop\compiler>
```

Not Accepted Test Cases

- Handled if the return type declared is anything except int , char, string, float.



The screenshot shows the Visual Studio Code interface with a C program named `sample.c` open. The code defines a public interface `dhaoqli` with a function `friends` that takes an integer and returns a `random_type`. The terminal shows the output of the compiler, which reports a syntax error for the `random_type` return type.

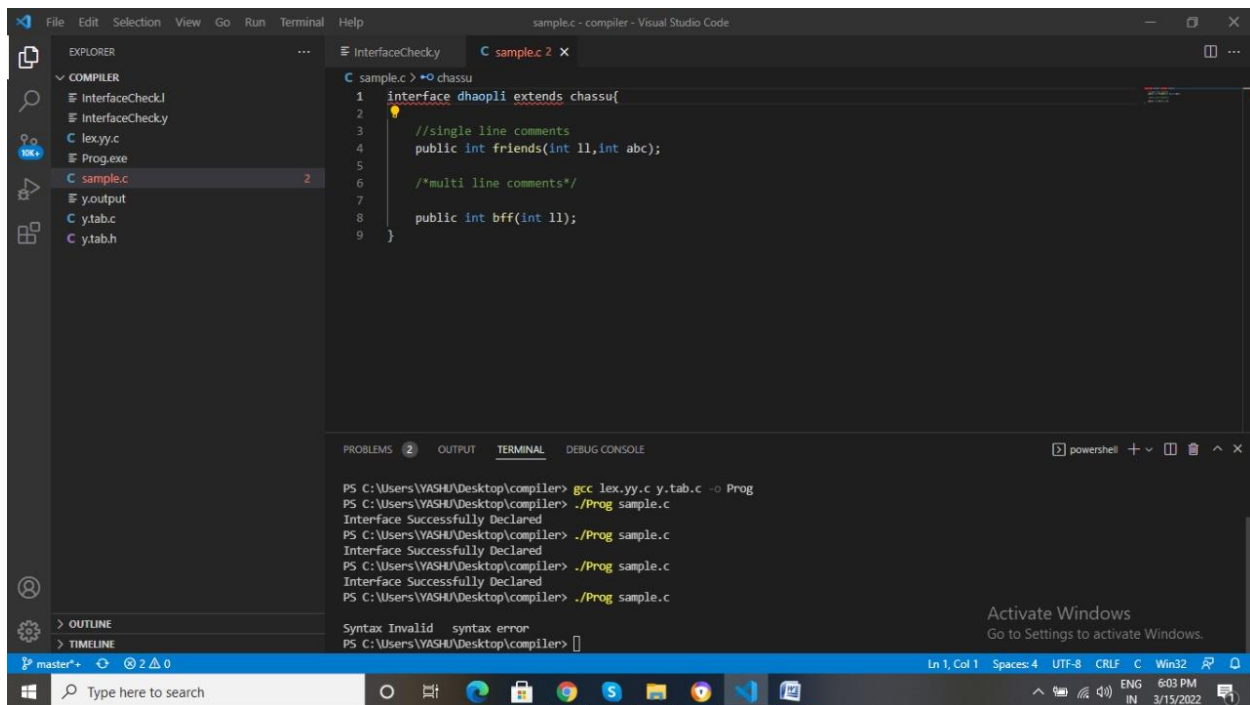
```
sample.c > dhaoqli > friends(int)
1 public interface dhaoqli {
2     public random_type friends(int a);
3
4 }
5 End of interface
```

```
Syntax Invalid syntax error
PS C:\Users\YASHU\Desktop\compiler> ./Prog sample.c

Syntax Invalid syntax error
PS C:\Users\YASHU\Desktop\compiler> ./Prog sample.c
Interface Successfully Declared
PS C:\Users\YASHU\Desktop\compiler> ./Prog sample.c

Syntax Invalid syntax error
PS C:\Users\YASHU\Desktop\compiler>
```

- Handled if public keyword is not written before interface keyword.



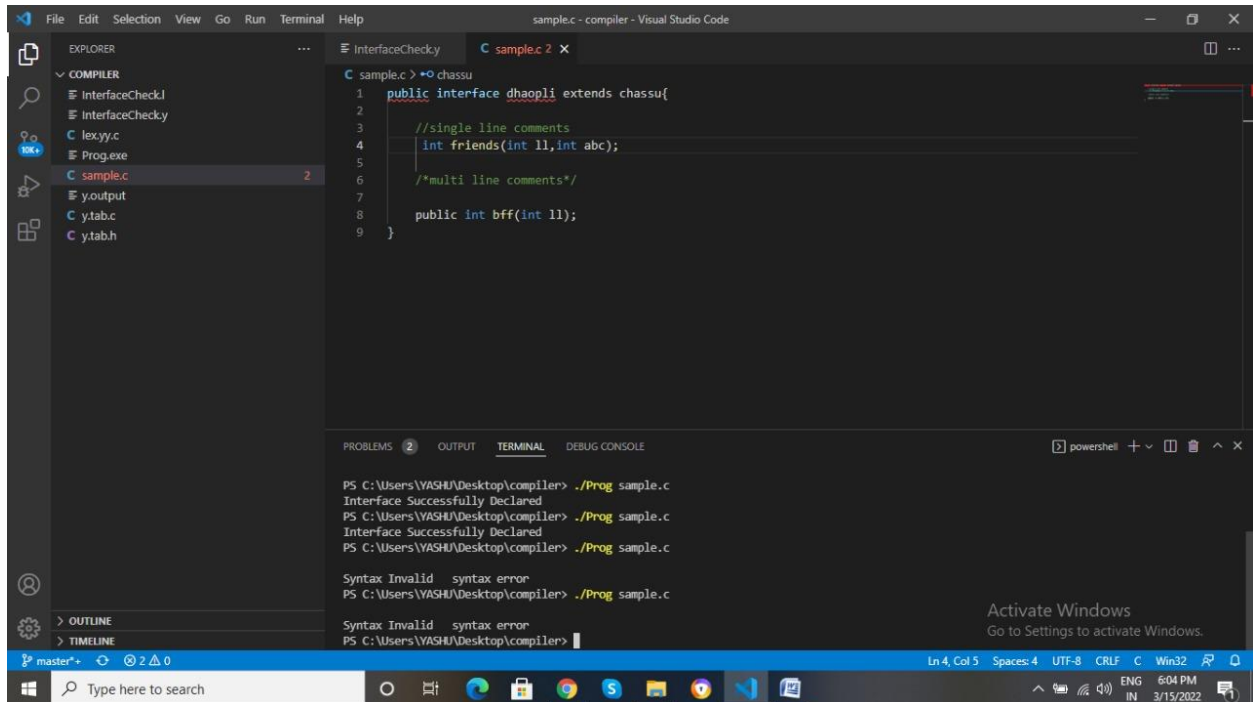
The screenshot shows the Visual Studio Code interface with a C program named `sample.c` open. The code defines an interface `dhaoqli` with a function `friends` that takes two integers and returns an integer. The terminal shows the output of the compiler, which reports a syntax error for the `dhaoqli` keyword.

```
sample.c > dhaoqli extends chassu{
1 interface dhaoqli extends chassu{
2
3     //single line comments
4     public int friends(int ll,int abc);
5
6     /*multi line comments*/
7
8     public int bff(int ll);
9 }
```

```
PS C:\Users\YASHU\Desktop\compiler> gcc lex.yy.c y.tab.c -o Prog
PS C:\Users\YASHU\Desktop\compiler> ./Prog sample.c
Interface Successfully Declared
PS C:\Users\YASHU\Desktop\compiler> ./Prog sample.c
Interface Successfully Declared
PS C:\Users\YASHU\Desktop\compiler> ./Prog sample.c
Interface Successfully Declared
PS C:\Users\YASHU\Desktop\compiler> ./Prog sample.c

Syntax Invalid syntax error
PS C:\Users\YASHU\Desktop\compiler>
```

7. Handled if we miss return data type.



The screenshot shows the Visual Studio Code editor with a C# file named `sample.c` open. The code defines an interface `dhaopli` that extends `chassu`. It includes a single-line comment, a function `friends` with parameters `(int ll, int abc)`, a multi-line comment, and a function `bff` with parameter `(int ll)`. The terminal shows the output of the command `./Prog sample.c`, which successfully declares the interface. However, the compiler reports syntax errors for the `friends` and `bff` functions, indicating that the return data type is missing.

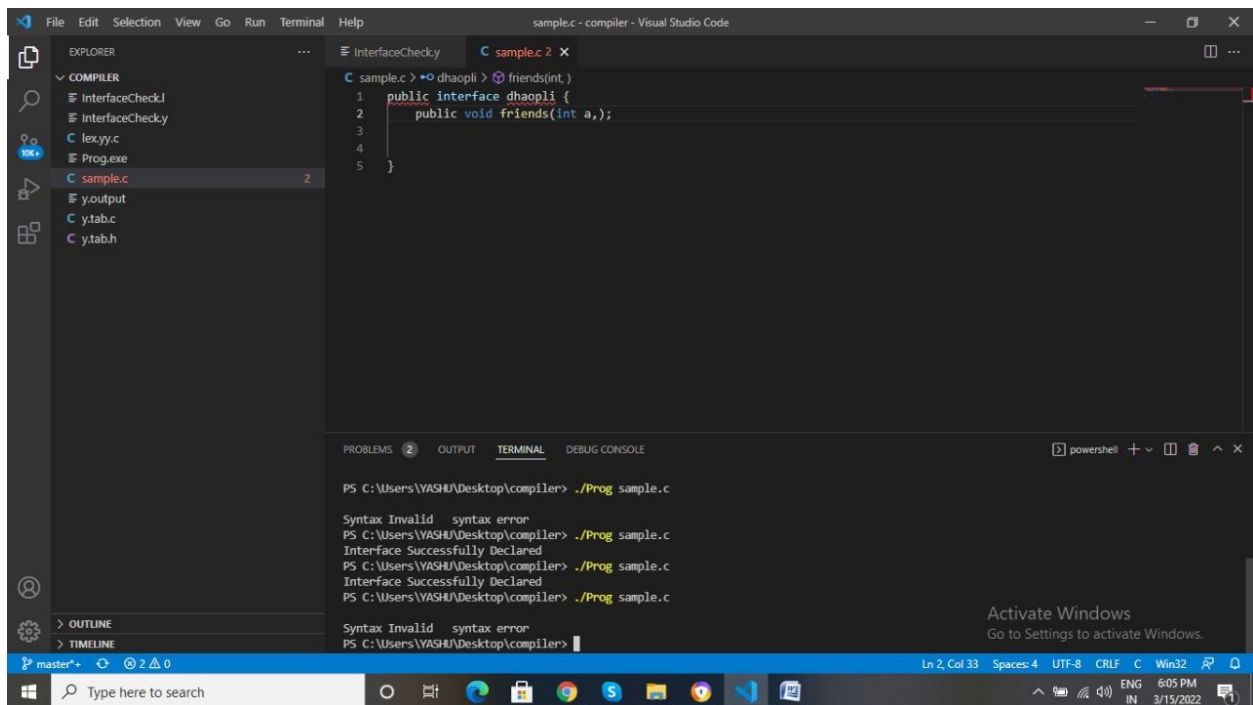
```
1 public interface dhaopli extends chassu{
2
3     //single line comments
4     int friends(int ll,int abc);
5
6     /*multi line comments*/
7
8     public int bff(int ll);
9 }
```

```
PS C:\Users\YASHU\Desktop\compiler> ./Prog sample.c
Interface Successfully Declared
PS C:\Users\YASHU\Desktop\compiler> ./Prog sample.c
Interface Successfully Declared
PS C:\Users\YASHU\Desktop\compiler> ./Prog sample.c

Syntax Invalid syntax error
PS C:\Users\YASHU\Desktop\compiler> ./Prog sample.c

Syntax Invalid syntax error
PS C:\Users\YASHU\Desktop\compiler>
```

8. Handled if we put “,” after function parameters.



The screenshot shows the Visual Studio Code editor with a C# file named `sample.c` open. The code defines an interface `dhaopli` that extends `chassu`. It includes a function `friends` with parameters `(int a,)`. The terminal shows the output of the command `./Prog sample.c`, which successfully declares the interface. However, the compiler reports a syntax error for the `friends` function, indicating that a comma is present after the function parameters.

```
1 public interface dhaopli {
2     public void friends(int a,);
3
4
5 }
```

```
PS C:\Users\YASHU\Desktop\compiler> ./Prog sample.c

Syntax Invalid syntax error
PS C:\Users\YASHU\Desktop\compiler> ./Prog sample.c
Interface Successfully Declared
PS C:\Users\YASHU\Desktop\compiler> ./Prog sample.c
Interface Successfully Declared
PS C:\Users\YASHU\Desktop\compiler> ./Prog sample.c

Syntax Invalid syntax error
PS C:\Users\YASHU\Desktop\compiler>
```