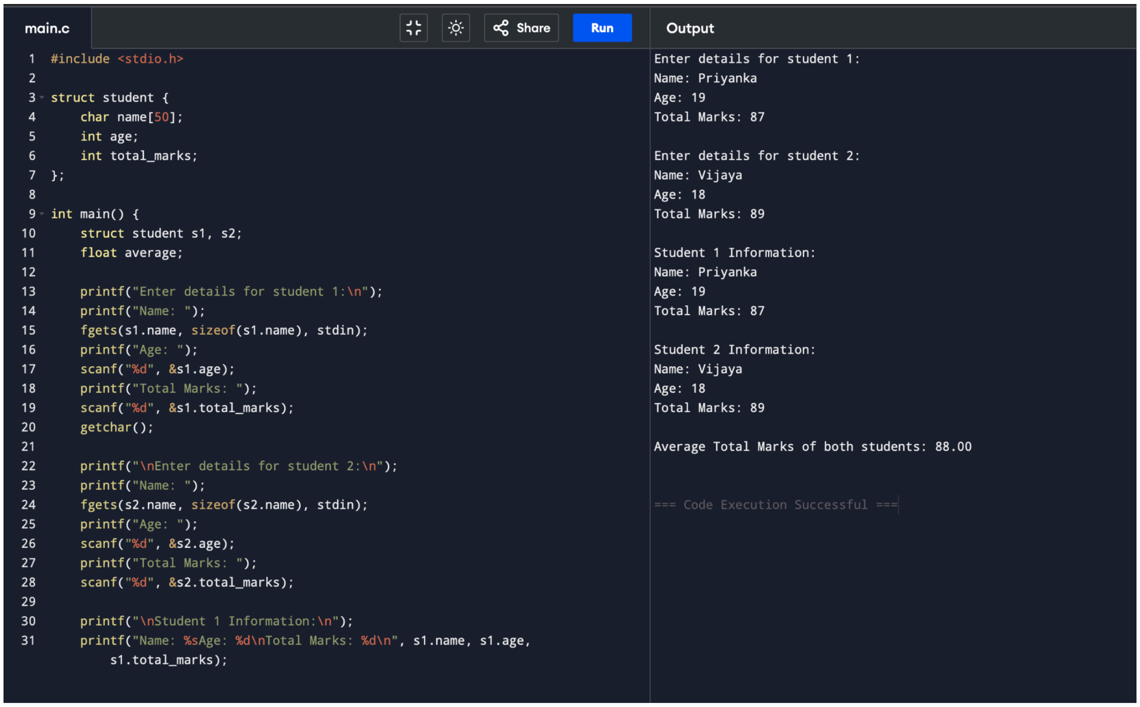
LAB ASSIGNMENT 2

1.Create a structure called “Student” with members name,age and total marks.Write a C program to input data for two students,display their information and find the average of total marks



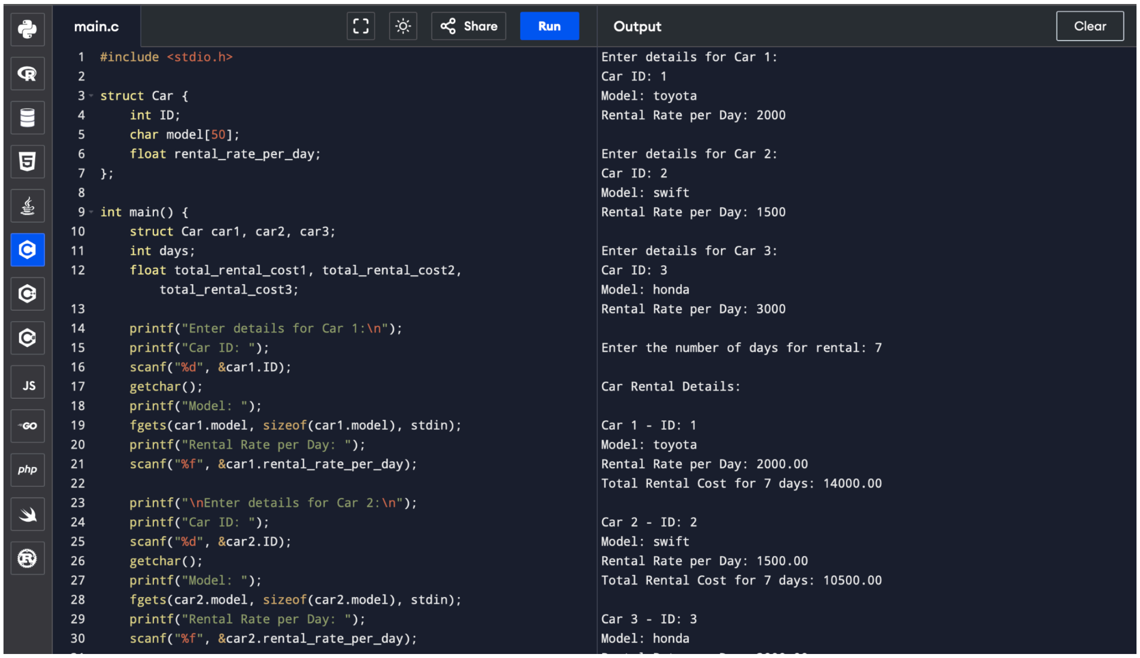
```
#include <stdio.h>

struct student {
    char name[50];
    int age;
    int total_marks;
};

int main() {
    struct student s1, s2;
    float average;

    printf("Enter details for student 1:\n");
    printf("Name: ");
    fgets(s1.name, sizeof(s1.name), stdin);
    printf("Age: ");
    scanf("%d", &s1.age);
    printf("Total Marks: ");
    scanf("%d", &s1.total_marks);
    getchar();

    printf("\nEnter details for student 2:\n");
    printf("Name: ");
    fgets(s2.name, sizeof(s2.name), stdin);
    printf("Age: ");
    scanf("%d", &s2.age);
    printf("Total Marks: ");
    scanf("%d", &s2.total_marks);

    printf("\nStudent 1 Information:\n");
    printf("Name: %sAge: %d\nTotal Marks: %d\n", s1.name, s1.age,
        s1.total_marks);
```

```
Enter details for student 1:
Name: Priyanka
Age: 19
Total Marks: 87

Enter details for student 2:
Name: Vijaya
Age: 18
Total Marks: 89

Student 1 Information:
Name: Priyanka
Age: 19
Total Marks: 87

Student 2 Information:
Name: Vijaya
Age: 18
Total Marks: 89

Average Total Marks of both students: 88.00


=== Code Execution Successful ===
```

```
#include <stdio.h>

struct student {
    char name[50];
    int age;
    int total_marks;
};

int main() {
    struct student s1, s2;
    float average;

    printf("Enter details for student 1:\n");
    printf("Name: ");
    fgets(s1.name, sizeof(s1.name), stdin);
    printf("Age: ");
    scanf("%d", &s1.age);
    printf("Total Marks: ");
    scanf("%d", &s1.total_marks);
    getchar();
    printf("\nEnter details for student 2:\n");
    printf("Name: ");
    fgets(s2.name, sizeof(s2.name), stdin);
    printf("Age: ");
    scanf("%d", &s2.age);
    printf("Total Marks: ");
    scanf("%d", &s2.total_marks);

    printf("\nStudent 1 Information:\n");
    printf("Name: %sAge: %d\nTotal Marks: %d\n", s1.name, s1.age, s1.total_marks);

    printf("\nStudent 2 Information:\n");
    printf("Name: %sAge: %d\nTotal Marks: %d\n", s2.name, s2.age, s2.total_marks);

    average = (s1.total_marks + s2.total_marks) / 2.0;

    printf("\nAverage Total Marks of both students: %.2f\n", average);

    return 0;
}
```

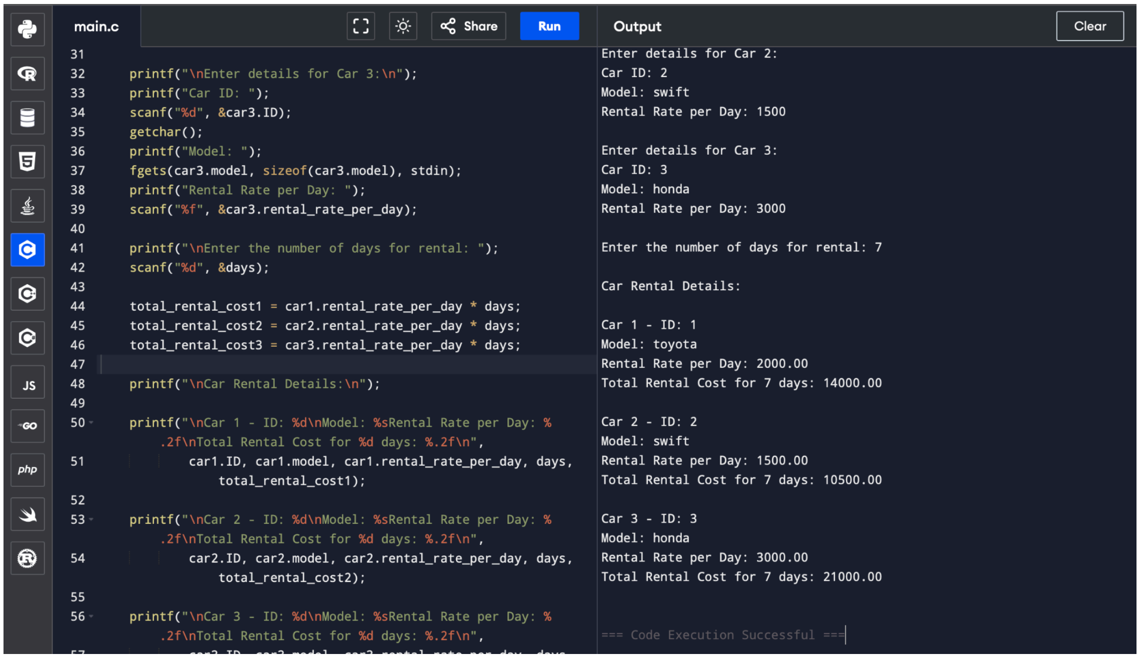
2.Design a structure named “Car” to store details like ID,model and rental rate per day.Write a C program to input data for three cars,calculate the total rental cost for a specified number of days,and display the results.

```
#include <stdio.h>

// Structure definition
struct Car {
    int ID;
    char model[50];
    float rental_rate_per_day;
};

int main() {
    struct Car car1, car2, car3;
    int days;
    float total_rental_cost1, total_rental_cost2, total_rental_cost3
    printf("Enter details for Car 1:\n");
    printf("Car ID: ");
    scanf("%d", &car1.ID);
    getchar();
    printf("Model: ");
    fgets(car1.model, sizeof(car1.model), stdin);
    printf("Rental Rate per Day: ");
    scanf("%f", &car1.rental_rate_per_day);

    printf("\nEnter details for Car 2:\n");
    printf("Car ID: ");
    scanf("%d", &car2.ID);
    getchar();
    printf("Model: ");
    fgets(car2.model, sizeof(car2.model), stdin);
    printf("Rental Rate per Day: ");
    scanf("%f", &car2.rental_rate_per_day);

    printf("\nEnter details for Car 3:\n");
    printf("Car ID: ");
    scanf("%d", &car3.ID);
    getchar();
    printf("Model: ");
    fgets(car3.model, sizeof(car3.model), stdin);
    printf("Rental Rate per Day: ");
    scanf("%f", &car3.rental_rate_per_day)
    printf("\nEnter the number of days for rental: ");
    scanf("%d", &days);

    total_rental_cost1 = car1.rental_rate_per_day * days;
    total_rental_cost2 = car2.rental_rate_per_day * days;
    total_rental_cost3 = car3.rental_rate_per_day * days;
```



```c
#include <stdio.h>

struct Car {
    int ID;
    char model[50];
    float rental_rate_per_day;
};

int main() {
    struct Car car1, car2, car3;
    int days;
    float total_rental_cost1, total_rental_cost2,
        total_rental_cost3;

    printf("Enter details for Car 1:\n");
    printf("Car ID: ");
    scanf("%d", &car1.ID);
    getchar();
    printf("Model: ");
    fgets(car1.model, sizeof(car1.model), stdin);
    printf("Rental Rate per Day: ");
    scanf("%f", &car1.rental_rate_per_day);

    printf("\nEnter details for Car 2:\n");
    printf("Car ID: ");
    scanf("%d", &car2.ID);
    getchar();
    printf("Model: ");
    fgets(car2.model, sizeof(car2.model), stdin);
    printf("Rental Rate per Day: ");
    scanf("%f", &car2.rental_rate_per_day);
```

```
Enter details for Car 1:
Car ID: 1
Model: toyota
Rental Rate per Day: 2000

Enter details for Car 2:
Car ID: 2
Model: swift
Rental Rate per Day: 1500

Enter details for Car 3:
Car ID: 3
Model: honda
Rental Rate per Day: 3000

Enter the number of days for rental: 7

Car Rental Details:

Car 1 - ID: 1
Model: toyota
Rental Rate per Day: 2000.00
Total Rental Cost for 7 days: 14000.00

Car 2 - ID: 2
Model: swift
Rental Rate per Day: 1500.00
Total Rental Cost for 7 days: 10500.00

Car 3 - ID: 3
Model: honda
```



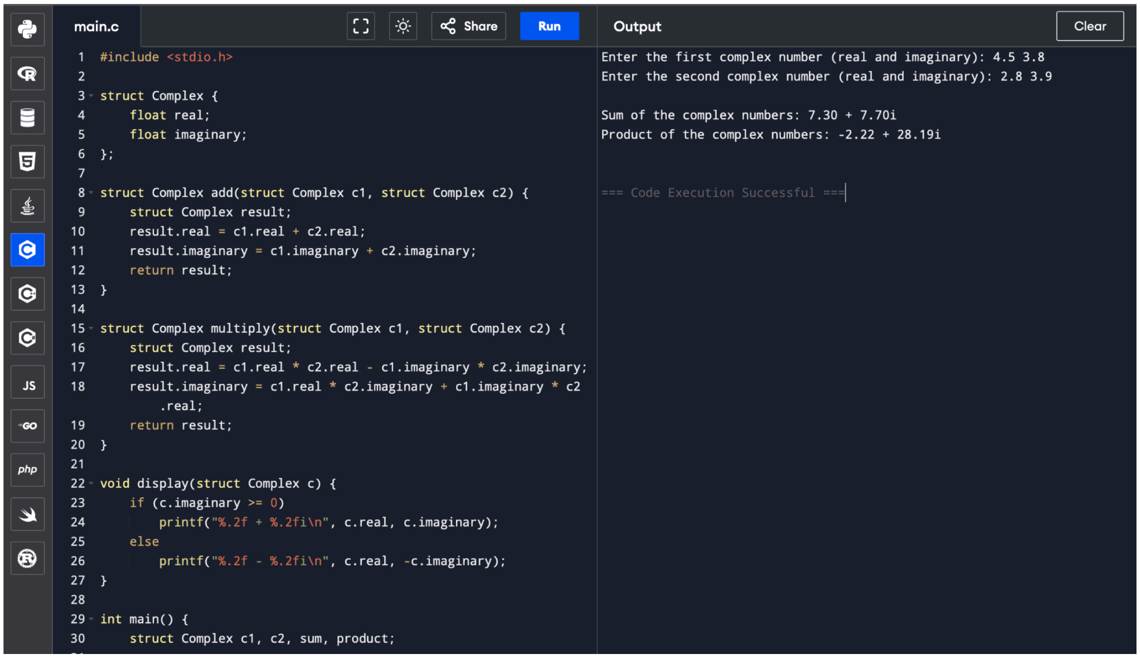
```c

    printf("\nEnter details for Car 3:\n");
    printf("Car ID: ");
    scanf("%d", &car3.ID);
    getchar();
    printf("Model: ");
    fgets(car3.model, sizeof(car3.model), stdin);
    printf("Rental Rate per Day: ");
    scanf("%f", &car3.rental_rate_per_day);

    printf("\nEnter the number of days for rental: ");
    scanf("%d", &days);

    total_rental_cost1 = car1.rental_rate_per_day * days;
    total_rental_cost2 = car2.rental_rate_per_day * days;
    total_rental_cost3 = car3.rental_rate_per_day * days;

    printf("\nCar Rental Details:\n");

    printf("\nCar 1 - ID: %d\nModel: %sRental Rate per Day: %
        .2f\nTotal Rental Cost for %d days: %.2f\n",
            car1.ID, car1.model, car1.rental_rate_per_day, days,
                total_rental_cost1);

    printf("\nCar 2 - ID: %d\nModel: %sRental Rate per Day: %
        .2f\nTotal Rental Cost for %d days: %.2f\n",
            car2.ID, car2.model, car2.rental_rate_per_day, days,
                total_rental_cost2);

    printf("\nCar 3 - ID: %d\nModel: %sRental Rate per Day: %
        .2f\nTotal Rental Cost for %d days: %.2f\n",
```

```
Enter details for Car 2:
Car ID: 2
Model: swift
Rental Rate per Day: 1500

Enter details for Car 3:
Car ID: 3
Model: honda
Rental Rate per Day: 3000

Enter the number of days for rental: 7

Car Rental Details:

Car 1 - ID: 1
Model: toyota
Rental Rate per Day: 2000.00
Total Rental Cost for 7 days: 14000.00

Car 2 - ID: 2
Model: swift
Rental Rate per Day: 1500.00
Total Rental Cost for 7 days: 10500.00

Car 3 - ID: 3
Model: honda
Rental Rate per Day: 3000.00
Total Rental Cost for 7 days: 21000.00


=== Code Execution Successful ===
```

```c
    printf("\nCar Rental Details:\n");

    printf("\nCar 1 - ID: %d\nModel: %sRental Rate per Day: %.2f\nTotal Rental Cost for %d days: %.2f\n",
            car1.ID, car1.model, car1.rental_rate_per_day, days, total_rental_cost1);

    printf("\nCar 2 - ID: %d\nModel: %sRental Rate per Day: %.2f\nTotal Rental Cost for %d days: %.2f\n",
            car2.ID, car2.model, car2.rental_rate_per_day, days, total_rental_cost2);
    printf("\nCar 3 - ID: %d\nModel: %sRental Rate per Day: %.2f\nTotal Rental Cost for %d
days: %.2f\n",
            car3.ID, car3.model, car3.rental_rate_per_day, days, total_rental_cost3);

    return 0;
}
```

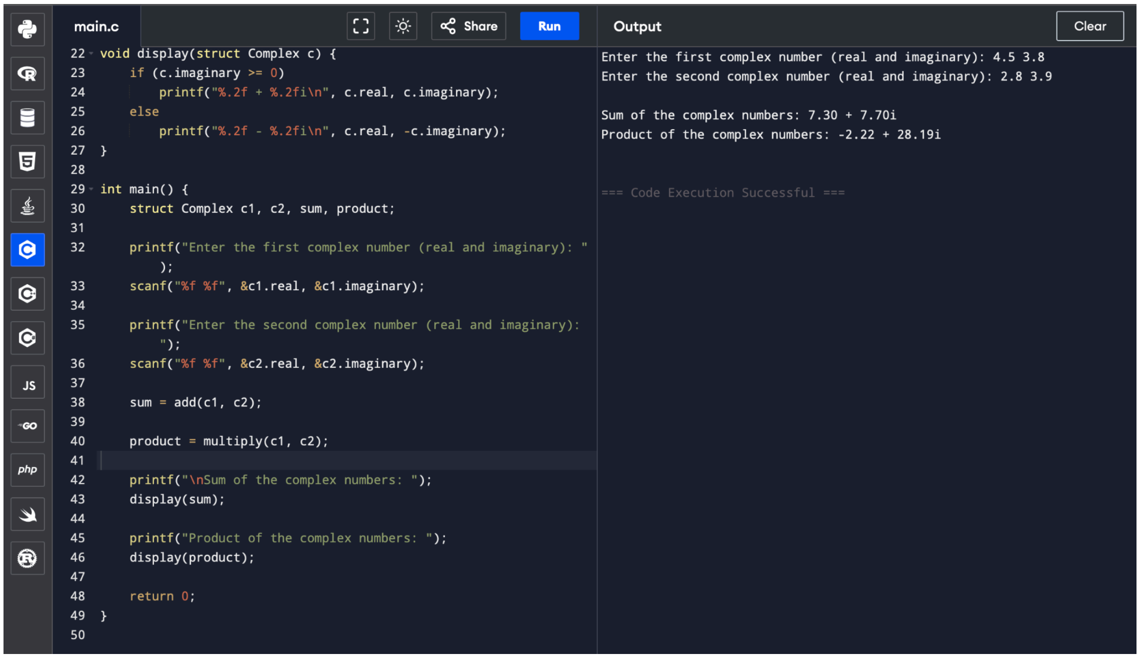
3.Create a structure named Complex to represent a complex number with real and imaginary parts.Write a C program to add and multiply two complex numbers

```
#include <stdio.h>

struct Complex {
    float real;
    float imaginary;
};

struct Complex add(struct Complex c1, struct Complex c2) {
    struct Complex result;
    result.real = c1.real + c2.real;
    result.imaginary = c1.imaginary + c2.imaginary;
    return result;
}

struct Complex multiply(struct Complex c1, struct Complex c2) {
    struct Complex result;
    result.real = c1.real * c2.real - c1.imaginary * c2.imaginary;
    result.imaginary = c1.real * c2.imaginary + c1.imaginary * c2.real;
    return result;
}

void display(struct Complex c) {
    if (c.imaginary >= 0)
        printf("%.2f + %.2fi\n", c.real, c.imaginary);
    else
        printf("%.2f - %.2fi\n", c.real, -c.imaginary);
}

int main() {
    struct Complex c1, c2, sum, product;

    // Input complex numbers
    printf("Enter the first complex number (real and imaginary): ");
    scanf("%f %f", &c1.real, &c1.imaginary);

    printf("Enter the second complex number (real and imaginary): ");
    scanf("%f %f", &c2.real, &c2.imaginary);

    sum = add(c1, c2);
    product = multiply(c1, c2);
```



```c
#include <stdio.h>

struct Complex {
    float real;
    float imaginary;
};

struct Complex add(struct Complex c1, struct Complex c2) {
    struct Complex result;
    result.real = c1.real + c2.real;
    result.imaginary = c1.imaginary + c2.imaginary;
    return result;
}

struct Complex multiply(struct Complex c1, struct Complex c2) {
    struct Complex result;
    result.real = c1.real * c2.real - c1.imaginary * c2.imaginary;
    result.imaginary = c1.real * c2.imaginary + c1.imaginary * c2
        .real;
    return result;
}

void display(struct Complex c) {
    if (c.imaginary >= 0)
        printf("%.2f + %.2fi\n", c.real, c.imaginary);
    else
        printf("%.2f - %.2fi\n", c.real, -c.imaginary);
}

int main() {
    struct Complex c1, c2, sum, product;
```

```
Enter the first complex number (real and imaginary): 4.5 3.8
Enter the second complex number (real and imaginary): 2.8 3.9

Sum of the complex numbers: 7.30 + 7.70i
Product of the complex numbers: -2.22 + 28.19i


=== Code Execution Successful ===
```



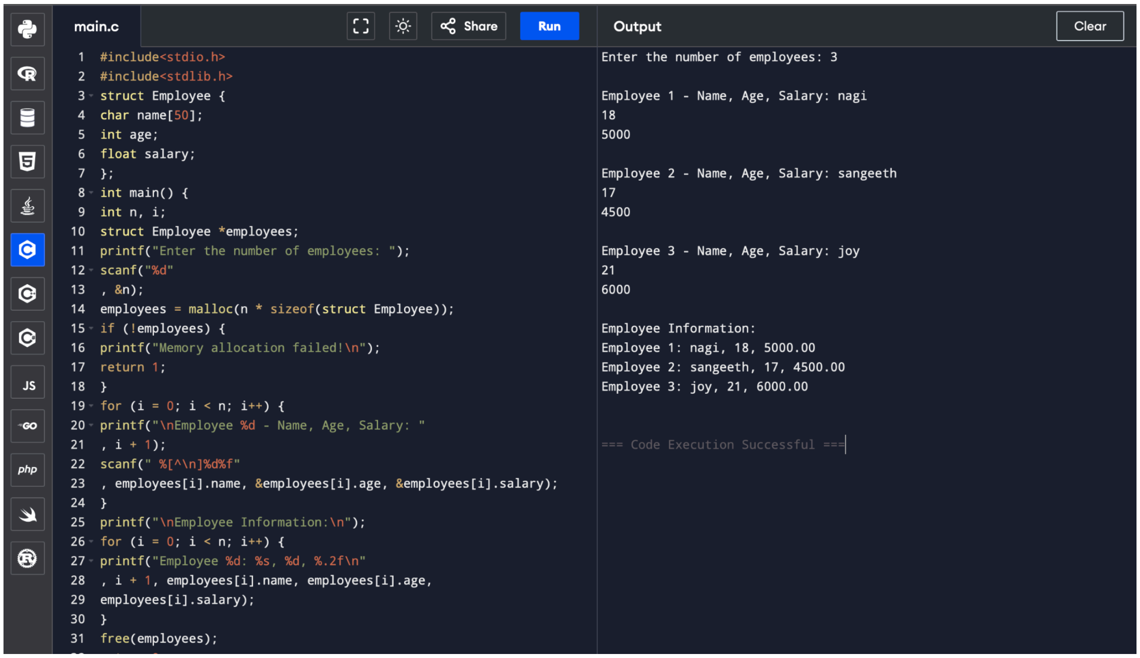
```c
void display(struct Complex c) {
    if (c.imaginary >= 0)
        printf("%.2f + %.2fi\n", c.real, c.imaginary);
    else
        printf("%.2f - %.2fi\n", c.real, -c.imaginary);
}

int main() {
    struct Complex c1, c2, sum, product;

    printf("Enter the first complex number (real and imaginary): "
        );
    scanf("%f %f", &c1.real, &c1.imaginary);

    printf("Enter the second complex number (real and imaginary):
        ");
    scanf("%f %f", &c2.real, &c2.imaginary);

    sum = add(c1, c2);

    product = multiply(c1, c2);

    printf("\nSum of the complex numbers: ");
    display(sum);

    printf("Product of the complex numbers: ");
    display(product);

    return 0;
}

```

```
Enter the first complex number (real and imaginary): 4.5 3.8
Enter the second complex number (real and imaginary): 2.8 3.9

Sum of the complex numbers: 7.30 + 7.70i
Product of the complex numbers: -2.22 + 28.19i


=== Code Execution Successful ===
```

```
    printf("\nSum of the complex numbers: ");
    display(sum);

    printf("Product of the complex numbers: ");
    display(product);

    return 0;
}
```

4.Write a program in C to store and print the information of N employees using dynamic memory allocation and structures.

```
#include<stdio.h>
#include<stdlib.h>
struct Employee {
char name[50];
int age;
float salary;
};
int main() {
int n, i;
struct Employee *employees;
printf("Enter the number of employees: ");
scanf("%d"
, &n);
employees = malloc(n * sizeof(struct Employee));
if (!employees) {
printf("Memory allocation failed!\n");
return 1;
}
for (i = 0; i < n; i++) {
printf("\nEmployee %d - Name, Age, Salary: "
, i + 1);
scanf(" %[^\n]%d%f"
, employees[i].name, &employees[i].age, &employees[i].salary);
}
printf("\nEmployee Information:\n");
for (i = 0; i < n; i++) {
printf("Employee %d: %s, %d, %.2f\n"
, i + 1, employees[i].name, employees[i].age,
employees[i].salary);
}
free(employees);
return 0;
}
```



```
#include<stdio.h>
#include<stdlib.h>
struct Employee {
char name[50];
int age;
float salary;
};
int main() {
int n, i;
struct Employee *employees;
printf("Enter the number of employees: ");
scanf("%d"
, &n);
employees = malloc(n * sizeof(struct Employee));
if (!employees) {
printf("Memory allocation failed!\n");
return 1;
}
for (i = 0; i < n; i++) {
printf("\nEmployee %d - Name, Age, Salary: "
, i + 1);
scanf(" %[^\n]%d%f"
, employees[i].name, &employees[i].age, &employees[i].salary);
}
printf("\nEmployee Information:\n");
for (i = 0; i < n; i++) {
printf("Employee %d: %s, %d, %.2f\n"
, i + 1, employees[i].name, employees[i].age,
employees[i].salary);
}
free(employees);
```

```
Enter the number of employees: 3

Employee 1 - Name, Age, Salary: nagi
18
5000

Employee 2 - Name, Age, Salary: sangeeth
17
4500

Employee 3 - Name, Age, Salary: joy
21
6000

Employee Information:
Employee 1: nagi, 18, 5000.00
Employee 2: sangeeth, 17, 4500.00
Employee 3: joy, 21, 6000.00


=== Code Execution Successful ===
```