Rajiv-0920 now

664 lines (359 loc) · 5.4 KB

Preview Code Blame

Basic Simple C Programs

1. WAP(Write a program) to print "Hello World".

> Expected Output

```
Hello World
```

> Source Code

```c
#include <stdio.h>

int main(){
    printf("Hello World");
    return 0;
}
```

---

2. WAP. to input a number and print them.

> Test Data

```
Enter a number: 5
```

> Expected Output

```
Number is 5
```

> Source Code

```c
#include <stdio.h>

int main(){
    int num;
```

```
    printf("Enter a number");
    scanf("%d", &num);

    printf("Number is %d", num);

    return 0;
}
```

---

3. WAP. to accept two numbers and print their addition, subtraction, multiplication, division.

> Test Data

```
Enter 2 number: 5 10
```

> Expected Output

```
Addition = 15
Subtraction = -5
Multiplication = 50
Division = 0
```

> Source Code

```
#include <stdio.h>

int main(){

    int add, mul, sub, div, num1, num2;

    printf("\nEnter 2 number:");
    scanf("%d %d", &num1, &num2);

    add = num1 + num2;
    sub = num1 - num2;
    mul = num1 * num2;
    div = num1 / num2;

    printf("Addition = %d \nSubtraction = %d \nMultiplication = %d \nDivision = %d"
    , add, sub, mul, div);

    return 0;
}
```

---

4. WAP. to input a number and change the sign.

> Test Data

```
Enter a number: 5
```

```
Enter a number: -5
```

> Expected Output

```
Changed number = -5
Changed number = 5
```

> Source Code

```c
#include <stdio.h>

int main(){
    int num;

    printf("Enter a number: ");
    scanf("%d", &num);

    num = num * -1;

    printf("Changed number = %d", num);
    return 0;
}
```

---

5. WAP. to input two number and display quotient and remainder.

> Test Data

```
input number : 98 4
```

> Expected Output

```
quotient = 24
remainder = 2
```

> Source Code

```c
#include <stdio.h>

int main(){
    int dividend, divisor, quotient, remainder;

    printf("Enter Dividend and Division: ");
    scanf("%d %d", &dividend, &divisor);

    quotient = dividend / divisor;
    remainder = dividend % divisor;

    printf("Quotient = %d Remainder = %d", quotient, remainder);

    return 0;
}
```

6. WAP. to display last digit of a number.

> Test Data

```
Enter a number: 153
```

> Expected Output

```
Last digit = 3
```

> Source Code

```c
#include <stdio.h>

int main(){
    int num;

    printf("Enter a number: ");
    scanf("%d", &num);

    num = num % 10;

    printf("Last digit = %d", num);

    return 0;
}
```

7. WAP. to accept a number from user and print it's square & cube in C language.

> Test Data

```
Enter a number = 5
```

> Expected Output

```
Square = 25 Cube = 125
```

> Source Code

```c
#include <stdio.h>

int main(){

    int num, cube, square;

    printf("Enter a number: ");
    scanf("%d", &num);
```

```
    cube = num * num * num;
    square = num * num;

    printf("Square: %d Cube: %d \n", square, cube);

    return 0;
}
```

---

8. WAP. to calculate Area and Circumference of a Circle.

> Formula

Area of a Circle = πr^2

Circumference of a circle = $2\pi r$

> Test Data

```
Enter Radius: 15
```

> Expected Output

```
Area of a circle = 78.525002
Circumference of a circle = 31.410000
```

> Source Code

```
#include <stdio.h>

int main(){

    float area, circum, radius, pi = 3.14153;

    printf("\nEnter Radius: ");
    scanf("%f", &radius);

    area = radius * radius * pi;
    circum = 2 * pi * radius;

    printf("Area of the circle: %f \n", area);
    printf("Circumference of the circle: %f \n", circum);

    return 0;
}
```

9.

> Test Data

> Expected Output

> Source Code

4.

> Test Data
>
> Expected Output
>
> Source Code

4.

> Test Data
>
> Expected Output
>
> Source Code

4.

> Test Data
>
> Expected Output
>
> Source Code

4.

> Test Data
>
> Expected Output
>
> Source Code

4.

> Test Data

> Expected Output

> Source Code

```
```

4.

> Test Data

> Expected Output

> Source Code

```
```

4.

> Test Data

> Expected Output

> Source Code

```
```

4.

> Test Data

> Expected Output

> Source Code

```
```

4.

> Test Data

> Expected Output

> Source Code

```
```

4.

> Test Data

> Expected Output

> Source Code

```
```

4.

> Test Data

> Expected Output

> Source Code

```
```

4.

> Test Data

> Expected Output

> Source Code

```
```

4.

> Test Data

> Expected Output

> Source Code

```
```

4.

> Test Data

> Expected Output

> Source Code

4.

> Test Data
>
> Expected Output
>
> Source Code

4.

> Test Data
>
> Expected Output
>
> Source Code

4.

> Test Data
>
> Expected Output
>
> Source Code

4.

> Test Data
>
> Expected Output
>
> Source Code

4.

> Test Data

> Expected Output

> Source Code

4.

> Test Data

> Expected Output

> Source Code

4.

> Test Data

> Expected Output

> Source Code

4.

> Test Data

> Expected Output

> Source Code

4.

> Test Data

> Expected Output

> Source Code

4.

> Test Data

> Expected Output

> Source Code

4.

> Test Data

> Expected Output

> Source Code

4.

> Test Data

> Expected Output

> Source Code

4.

> Test Data

> Expected Output

> Source Code

4.

> Test Data

> Expected Output

> Source Code