```c
/*Getting started with C..................................................................
#Running a program in C...understanding the concept
=> Building a program
=> Run the file that gets build to execute
It will output the program in the console window
work on for loop, nested loops and 2D arrays
............comments.........
They are special block of codes which are usually ignored in a line of code
They can be used to disable temporarily lines of codes
They can be used to make notes
example of comments
1:/*Here you can type a line of code that will be ignored*/
2:// you can disable a line of code using this instead of deleting them
 //NB:well use them when necessary*/
//  NOTE EACH PROGRAM IS DONE ON A NEW FILE WITH A C EXTENSION
 //INTRO TO C...
#include <stdio.h>
#include <stdlib.h>

//Getting started with C programming......................................................
int main()
{
    printf("Hello world!\n");/* The semi-colon shows the instruction has ended to be executed*/
    printf("     /|\n");
    printf("    / |\n");
    printf("   /  |\n");
    printf("  /___|\n");
//Getting started with variable
//%s means insertion of a string
//%d means insertion of an integer

    char characterName[] = "Geestar"; //you can change the characters name here
    int CharacterAge = 18; // you can also modify the age
    printf("There once named a woman named %s\n", characterName); //%sThis is telling C that we want
to insert a collection of string
    printf("She was %d years old.\n", CharacterAge); //%d means we want to insert an integer
    printf("She really loved the name %s\n", characterName);
    printf("but did not like being %d.\n", CharacterAge);
    return 0;
}

//DATA TYPES..............................................................................
#include <stdio.h>
#include <stdlib.h>

int main()
{
    //working with data types
    int age = 40;
    double gpa = 3.6;
    char grade = 'A';
    char phrase[] = "Dee Xperts";

    return 0;
}

//Getting started with printf.............................................................
#include <stdio.h>
#include <stdlib.h>

int main()
{
    printf("Dee\nXperts\n");//\n is used to make a new line
    printf("My favorite number is %d\n", 8);// printing a format specifier
    printf("My favorite %s is %d\n", "Number", 8);//%s is used to insert a string and %d is used to
insert an integer.
    printf("My Favourite number is %f\n", 8.0);//%f is used to print a decimal number
    int favNum = 80;//We can as well create a variable and access it using the format specifier
    printf("My Favorite %s is %d\n", "Number", favNum);
    int favYear = 1998;
    printf("%s is %d\n", "My Favorite year is", favYear);
    char myChar = 'D';
    printf("My favorite letter is %c\n", myChar);//%c is used to show a single character
    return 0;
}

//Getting started with Numbers............................................................
#include <stdio.h>
#include <stdlib.h>
#include <math.h>

int main()
{
    printf("%f\n", 8.78);
    printf("%f\n", 4.5 * 9.7);//%f working with only a floating point number
    //working with variables
    int num = 76;
    printf("%d\n", num);
    //Working with functions
    printf("%f\n", pow(2, 3));//we use math.h to insert some of complex math functions
    printf("%f\n", sqrt(1024));
    printf("%f\n", ceil(36.758));//we use ceil to find the next highest number
    printf("%f\n", floor(36.758));// we use floor to do the opposite of ceil
return 0;
}

//Getting started with Constants..........................................................
#include <stdio.h>
#include <stdlib.h>

int main()
{
/*constants are special types of variables/value that cannot be modified*/
    int num = 5;
    printf("%d\n", num);
    num = 8;
    printf("%d\n", num);
//creating values that cannot be modified by using const
    const int num1 = 10;
    printf("%d\n", num1);
/*    num1 = 22;
    printf("%d\n", num1);*/ //if you proceed with code line 14 and 15 there will be an error because
you are trying to modify the variable/value
    return 0;
}

//Getting started with user_inputs........................................................
#include <stdio.h>
#include <stdlib.h>

int main()
{
//prompting the user
    int age;
    printf("Enter your age: ");
    scanf("%d", &age);// Allows the user to enter some info & >is a pointer and must be used when
scanf is used
    printf("You are %d years old\n", age);

    double gpa;
    printf("Enter your gpa: ");
    scanf("%lf", &gpa);
    printf("Your gpa is %f\n", gpa);

    char grade;
    printf("Enter your Grade: ");
    scanf("%c", &grade);
    printf("Your Grade is %c\n", grade);
    //getting string of characters
    char name [20];
    printf("Enter your name: ");
    fgets(name,20, stdin);//fgets prints the whole line of input from the user
    //scanf("%s", name);// scanf prints just a string if there is space between then it wont be printed
    printf("Your name is %s\n", name);
    return 0;
}

//Building a basic calculator............................................................
#include <stdio.h>
#include <stdlib.h>

int main()
{
    int num1;
    int num2;
    printf("Enter first number: ");
    scanf("%d", &num1);
    printf("Enter second number: ");
    scanf("%d", &num2);

    printf("Answer: %d\n", num1 + num2);
    /*The problem with the above code is that you cannot get an exact answer when using decimal*/

    double num3;
    double num4;
    printf("Enter the first number: ");
    scanf("%lf", &num3);
    printf("Enter the second number: ");
    scanf("%lf", &num4);

    printf("Answer: %f\n", num3 + num4);
    return 0;
}

//Getting started with the madlibs Game...................................................
#include <stdio.h>
#include <stdlib.h>

int main ()
{
//we going to create variables that are collections of character
    char color[20];
    char pluralNoun[20];
    char celebrityF[20]; //first name
    char celebrityL[20];// last name
//Here we going to prompt the user
    printf("Enter a color: ");
    scanf("%s", color); //in strings we do not nee the & sign in the scanf
    printf("Enter a plural noun: ");
    scanf("%s", pluralNoun);
    printf("Enter a celebrity: ");
    scanf("%s %s", celebrityF, celebrityL);//here we use them both because it cannot print out both
names

    printf("Roses are %s\n", color);
    printf("%s are blue\n", pluralNoun);
    printf("I love %s %s\n", celebrityF, celebrityL);

    return 0;
}

//Getting started with Arrays and Nested loops............................................
#include <stdio.h>
#include <stdlib.h>

int main()
{
//Arrays are used to store bunch of data values
//creating an array
    int luckyNumbers[] = {1, 2, 8, 15, 16, 23, 42};
    printf("%d\n", luckyNumbers[3]);//we can access an array using their index values[]
    //modifying the values in the array
    luckyNumbers[1] = 200;
    printf("%d\n", luckyNumbers[1]);
    int luckynumbers[10];
    luckynumbers[0] = 90;
    luckynumbers[1] = 200;
    printf("%d\n", luckynumbers[0]);
    printf("%d\n", luckynumbers[1]);

//2D Arrays
    int nums[3][2] = {
                    {1, 2},
                    {3, 4},
                    {5, 6}
                   };
    printf("%d", nums[1][1]);//Acessing the arrays
//Nested for loop
    int k, l;
    for(k = 0; k <3; k++)
    {
        for(l = 0; l < 2; l++)
        {
            printf("%d,", nums[k][l]);
        }
        printf("\n");
    }
    return 0;
}

//Getting started with Functions..........................................................
#include <stdio.h>
#include <stdlib.h>

int main ()
{
//collection of code that performs a specific task
    sayHi("Dunnieboy", 22);
    sayHi("Dunquins", 17);
    sayHi("GenG", 23);
    sayHi("Geestar", 18);
    return 0;
}

//creating a function
//using void means that we not going to return any info
//Naming a function with what it does
//calling the function in the main function above
void sayHi(char name [], int age){
    printf("Hello %s, you are %d years old\n", name, age); //note nothing is going to be displayed on the console window

}

//Getting started with Return statements.................................................
#include <stdio.h>
#include <stdlib.h>
//return type
//creating a function
//They should be created above the main function that is calling it

double cube(double num){
    double result = num * num * num;
    return result; //The return key breaks us out of the function
}
int main()
{
    printf("Answer: %f\n", cube(3.0));
    return 0;
}

//Getting started with If statements.....................................................
#include <stdio.h>
#include <stdlib.h>
/*Creating a function to determine the maximum number*/
//&& if used to check 2 conditions are true
//|| if or is used means that one of the condition is true
```

```c
int max(int num1, int num2, int num3){
    int result;
    if(num1 >= num2 && num1 >= num3 )
    {
        result = num1;
    }
    else if(num2 >= num1 && num2 >= num3)
    {
        result = num2;
    }
    else
    {
        result = num3;
    }
    return result;
}
int main()
{
    printf("%d\n", max(4, 10, 17));
    printf("\n");
//using or to check condition
    if(3 < 2 || 2 > 5)
    {
        printf("True\n");
    }
    else
    {
        printf("False\n");
    }
    if(3 == 2)
    {
        printf("True\n");
    }
    else
    {
        printf("False\n");
    }
    if(3 != 2)//this means is not
    {
        printf("True\n");
    }
    else
    {
        printf("False\n");
    }
    if(!(3 < 2)) //we use the exclamation sign when we want to negate something
    {
        printf("True\n");
    }
    else
    {
        printf("False\n");
    }
    return 0;
}
//Getting to bulild An Advanced calculator.............................................
#include <stdio.h>
#include <stdlib.h>
//Building an advanced calculator using if, else
int main()
{
    double num1;
    double num2;
    char op;

    printf("Enter a number: ");
    scanf("%lf", &num1);
    printf("Enter an operator: ");
    scanf(" %c", &op);
    printf("Enter another number: ");
    scanf("%lf", &num2);
    if(op == '+')
    {
        printf("%f", num1 + num2);
    }
    else if(op == '-')
    {
        printf("%f", num1 - num2);
    }
    else if(op == '/')
    {
        printf("%f", num1 / num2);
    }
    else if(op == '*')
    {
        printf("%f\n", num1 * num2);
    }
    else
    {
        printf("Invalid operator!\n");
    }
    return 0;
}

//Getting started with Swith and case.....................................................
#include <stdio.h>
#include <stdlib.h>
//comparison of one value with a bunch of other value
//creating a grade response
int main()
{
    char grade;
    printf("Enter your grade: ");
    scanf(" %c", &grade);

    switch(grade)
    {
        case 'A' :
            printf("You did great! \n");
            break; // break statements tells the switch statement to leave
        case 'B' :
            printf("You did alright! \n");
            break;
        case 'C' :
            printf("You did poorly! \n");
            break;
        case 'D' :
            printf("You did very bad! \n");
            break;
        case 'F' :
            printf("You Failed, TERRIBLY! \n");
            break;
        default :
            printf("Invalid grade\n");
    }
    return 0;
}

//Getting started ith struct..............................................................
#include <stdio.h>
#include <stdlib.h>
#include <string.h>//this is optional and was not part of the code

struct Student
{
    char name[50];
    int age;
    char major[50];
    double gpa;
};
int main()
{
```

```
//student1 is a container that will store the info of the student when the input their info
    struct Student student1;
    student1.age = 22;
    student1.gpa = 3.2;
    strcpy(student1.name, "Dunnieboy");
    strcpy(student1.major, "Computer science");

    printf("Your name is %s, you have majored in %s, you are %d years old, Your gpa is %f\n",
student1.name, student1.major, student1.age, student1.gpa);
    printf("\n");

    struct Student student2;
    student2.age = 18;
    student2.gpa = 3.0;
    strcpy(student2.name, "Geestar");
    strcpy(student2.major, "Programming");

    printf("Your name is %s, you have majored in %s, you are %d years old, Your gpa is %f\n",
student2.name, student2.major, student2.age, student2.gpa);
    return 0;
}

//Getting started with while loops.........................................................
#include <stdio.h>
#include <stdlib.h>
//The while loop is used to loop over and continuouly executing a code till a certain condition is
false
int main()
{
    int index = 1;
    while(index <= 10)//condition
    {
        printf("%d\n", index);
        index = index + 1;//we can instead use index++
    }
    printf(" There's a different code below\n");
//The Do while loops
    int a = 20;
    do
    {
        printf("%d\n", a);
        a++;//we can instead a = a + 1
    }
    while(a <= 100);

    return 0;
}

//Getting started with guessing Game......................................................
//Building a guessing game
#include <stdio.h>
#include <stdlib.h>

int main()
{
    int secretWord = 5;
    int guess;
    int guess_count = 0;
    int guess_limit = 5;
    int out_of_guesses = 0;

    while(guess != secretWord && out_of_guesses == 0)
    {
        if(guess_count < guess_limit)
        {
            printf("Enter your guess: ");
            scanf("%d", &guess);
            guess_count++;
        }

        else
        {
            out_of_guesses = 1;
        }
    }
    if(out_of_guesses == 1)
    {
        printf("You are out of guesses, You LOSE!\n");
    }
    else
     {
        printf("You win!\n");
     }
    /*int secretNumber = 5;
    int guess;

    while (guess != secretNumber)
    {
        printf("Enter a number: ");
        scanf("%d", &guess);
    }
    printf("You WIN!\n");*/
    return 0;
}

//Getting started with loop...............................................................
#include <stdio.h>
#include <stdlib.h>

int main()
{
    int j = 1;
    while(j <= 5)
    {
        printf("%d\n", j);
        j++;
    }
//The above while loop can be compressed to for loop below
    int i;
    for(i = 1; i < + 5; i++)
    {
        printf("%d\n", i);
    }
    return 0;
}

//Getting started with Memory_Address.....................................................
#include <stdio.h>
#include <stdlib.h>
//how computers store programs and where they are store
int main()
{
    int age = 30;
    double gpa = 3.4;
    char grade = 'A';

    printf("age: %p\ngpa: %p\ngrade: %p", &age, gpa, grade);//%p are pointer
    return 0;
}

//Getting started with pointers...........................................................
#include <stdio.h>
#include <stdlib.h>
//pointer must check on that
int main()
{
    int age = 30;
    int * pAge = &age;
    double gpa = 3.4;
    double *pGpa = &gpa;
    char grade = 'A';
    char *aGrade = &grade;

    printf("age's memory address: %p\n");
```

```c
    return 0;

    //Deferencing pointers check on that
}
//////////////////////////////////////////////////////////////////////////////////////////
/*Do as many example of coding as possible*/
```