




Programming Fundamentals

Aamina Batool



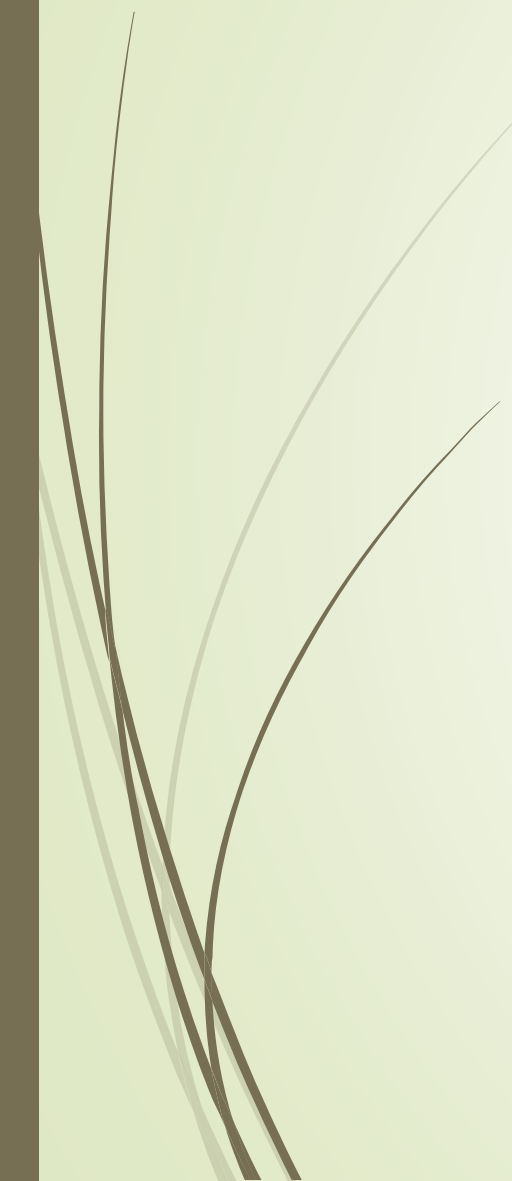
Nested for Loops




- for (init ; condition; update)
- { statement 1;
- statement 2;
- for (init; condition; update)
- { statement 1;
- statement 2;
- }
- statement 3;
- }



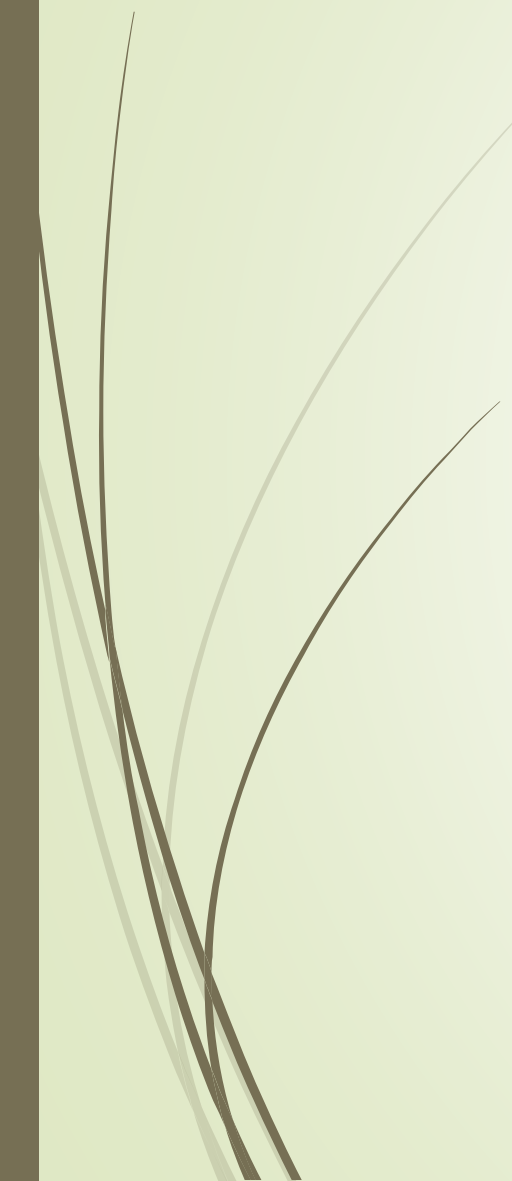
Nested while Loops





- while (condition)
- { statement 1;
- statement 2;
- while (condition)
- { statement 1;
- statement 2;
- }
- statement 3;
- }



Nested do-while Loops

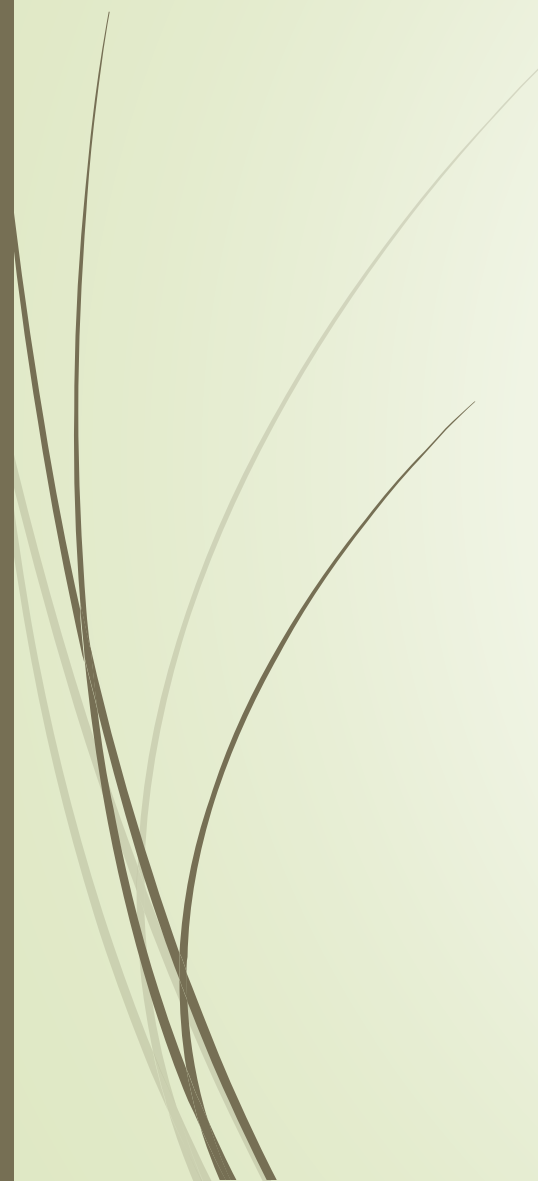
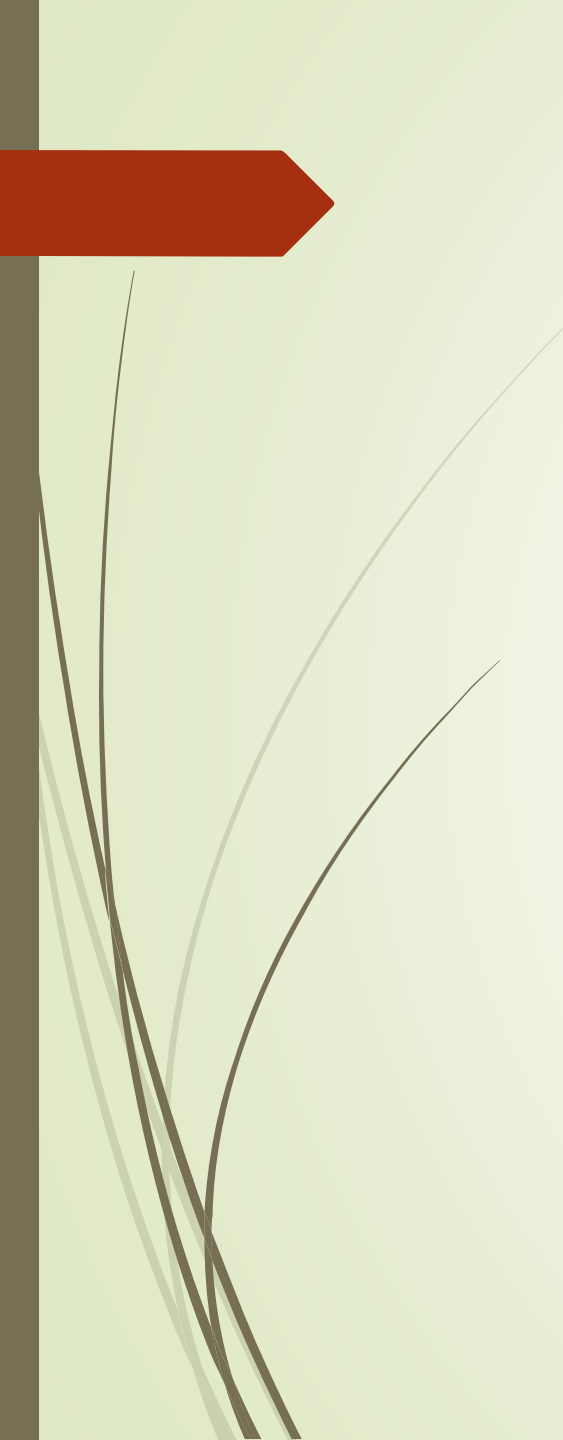


- do
- { statement 1;
- statement 2;
- do
- { statement 1;
- statement 2;
- } while (condition);
- statement 3;
- } while (condition);



```
int main ()
{
    int i = 0;



    while(i < 3)
    {
        int j = 0;
        while(j < 5)
        {
            cout << "i = " << i << " and j = " << j << endl;
            j++;
        }
        i++;
    }
    return 0;
}
```



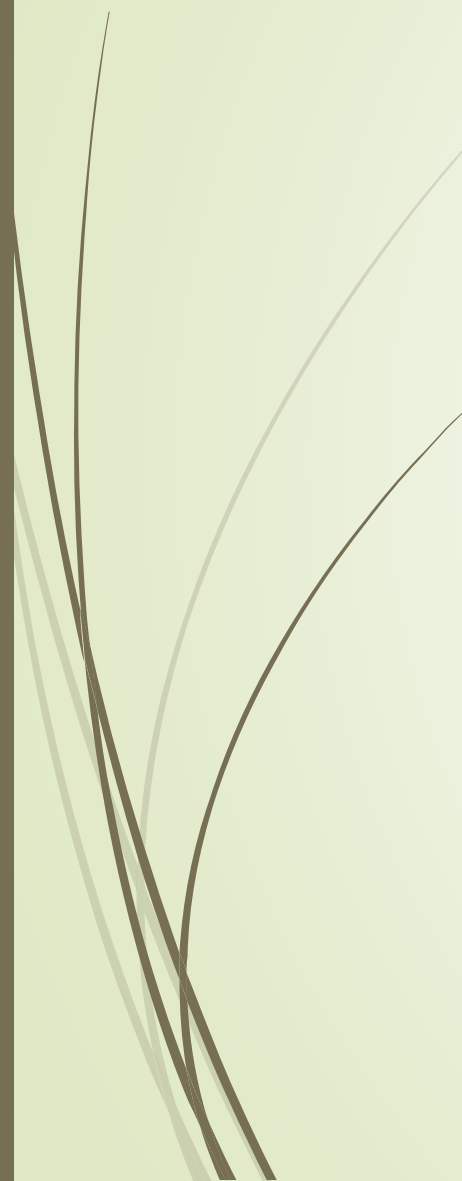

```
int main ()
{
    int i = 0;

    do
    {
        int j = 0;
        do
        {
            cout << "i = " << i << " and j = " << j << endl;
            j++;
        } while (j < 5);
        i++;
    } while (i < 3);

    return 0;
}
```



```
int main ()
{
    for(int i = 0; i < 3; i++)
    {
        int j = 0;
        for(int j = 0; j < 5; j++)
        {
            cout << "i = " << i << " and j = " << j << endl;
        }
    }
    return 0;
}
```



```
int main ()
{
    for(int i = 0; i < 3; i++)
    { cout << "Outer loop iteration #: " << i << endl;
      int j = 0;
      for(int j = 0; j < 5; j++)
      { cout << "Inner loop iteration #: " << j << endl;
        cout << "Sum = " << i + j << endl;
      }
      cout << "Inner loop completed" << endl;
      cout << "We are in outer loop" << endl;
    }
    return 0;
}
```




Nested Control Structures

```
#include <iostream>
using namespace std;

int main() {

    int rows = 5;
    int columns = 3;

    for (int i = 1; i <= rows; ++i) {
        for (int j = 1; j <= columns; ++j) {
            cout << "* ";
        }
        cout << endl;
    }

    return 0;
}
```

Nested Control Structures

```
for (i = 5; i >= 1 ; i--)  
{  
    for (j = 1; j <= i; j++)  
        cout << "*";  
    cout << endl;  
}
```

Nested Control Structures

```
for (i = 1; i <= 5 ; i++)  
{  
    for (j = 5; j >= i; j--)  
        cout << "*";  
    cout << endl;  
}
```

Break inside nested loops

```
#include <iostream>
using namespace std;

int main() {
    int weeks = 3, days_in_week = 7;

    for (int i = 1; i <= weeks; ++i) {
        cout << "Week: " << i << endl;

        for (int j = 1; j <= days_in_week; ++j) {
            // break during the 2nd week
            if (i == 2) {
                break;
            }
            cout << "    Day: " << j << endl;
        }
    }
}
```

Continue inside nested loops

```
#include <iostream>
using namespace std;

int main() {
    int weeks = 3, days_in_week = 7;

    for (int i = 1; i <= weeks; ++i) {
        cout << "Week: " << i << endl;

        for (int j = 1; j <= days_in_week; ++j) {
            // continue if the day is an odd number
            if (j % 2 != 0) {
                continue;
            }
            cout << "    Day: " << j << endl;
        }
    }
}
```