

-----guess output-----

```
static String str;
static DateTime time;

static void Main(string[] args)
{
    Console.WriteLine(str == null ? "str == null" : str);
    Console.WriteLine(time == null ? "time == null" : time.ToString());
    Console.ReadLine();
}
```

-----PrintHighestNumber-----

```
public static void PrintHighestNumber(int Position)
{
    int[] arr = { 11, 1, 3, 43, 54, 45, 46, 67 };
    int? max = arr.OrderByDescending().ElementAtOrDefault(Position - 1);
}
```

-----GetDataInDescending-----

```
int[] arr = new int[]{1,4,7,13,8,2};
int[] result = new int[arr.Length];
int prevMax = int.MaxValue;
for(int i = 0; i < arr.Length; i++){
    int max = arr[i];
    for(int j = i + 1; j < arr.Length; j++){
        if(arr[j] > max && arr[j] < prevMax){
            max = arr[j];
        }
    }
    result[i] = max;
    prevMax = max;
}
```

-----consecutive-----

```
/* Write Program to find consecutive character in string
*/
```

```
char[] charArray = { 'A', 'B', 'B', 'C', 'D', 'D', 'E', 'F', 'F' };
```

```
List<char> result = new List<char>();
char prevChar = charArray[0];
for(int i = 1; i < charArray.Length; i++){
    if(prevChar == charArray[i]){
        result.Add(charArray[i]);
    }
}
```

```

    }
    prevChar = charArray[i];
}

```

```

Console.WriteLine($"{string.Join(',', result)}");

```

-----character repetition count-----

Provide the C# code to print the character repetition count of a given string.

```

string str = "hellotherehowareyou";
Dictionary<char, int> charCount = new Dictionary<char, int>();

foreach (char c in str)
{
    if (charCount.ContainsKey(c))
    {
        charCount[c]++;
    }
    else
    {
        charCount[c] = 1;
    }
}

foreach (var pair in charCount)
{
    Console.WriteLine($"{pair.Key}: {pair.Value}");
}

```

--by linq

```

string str = "hellotherehowareyou";

var charCount = str
    .GroupBy(c => c)
    .Select(group => new { Character = group.Key, Count =
group.Count() });

foreach (var item in charCount)
{
    Console.WriteLine($"{item.Character}: {item.Count}");
}

```

----- Check if a Number is Palindrome-----

```

static bool IsPalindrome(int num)
{
    int originalNum = num;
    int reversedNum = 0;

    while (num > 0)
    {

```

```

        int digit = num % 10;
        reversedNum = reversedNum * 10 + digit;
        num /= 10;
    }

    return originalNum == reversedNum;
}

```

-----  
<https://www.toptal.com/c-sharp/interview-questions>

-----sql query-----

Employee - EmpId, Name  
 Department - DeptId, DeptName  
 EmployeeDepartment - Id, EmpId, DeptId

```

select count(*) DeptCount, isnull( d.DeptName, 'No Dept.') from EmployeeDepartment
ed join Department d on d.DeptId = ed.DeptId
where d.DeptName like 'a%'
group by d.DeptName having count(*)>5

```

t1=emp(nm,sal)

```

select top 1 sal
(
select distinct top 3 sal from emp order by sal desc
) as result
order by sal

```

select nm,sal from emp where sal = (

select distinct sal from emp order by sal desc offset 2 rows fetch next 1 rows  
 only

)

```

row_number = 1,2,3
rank() = 1,2,3,3,5
dense_rank() = 1, 2, 3, 3, 4, 5

```

A= 1,2,3,4,5

B= 0,1,2,3,4

I j = 1,2,3,4

L j = 1,2,3,4,5

R j = 0,1,2,3,4

```
c r = 25rows
```

```
select name from student where scores =(  
select max(scores) highestscores from student  
)
```

```
with cteresult as (  
    select *, dense_rank() over(partition by department order by scores) as  
    scoreRank from student  
)
```

```
select top 1 scores from (  
select distinct top 2 scores from student group by department order by scores desc  
) as tempStu order by scores
```

```
100=1  
100=1  
100=1  
200=2
```

```
select name from cteresult where scoreRank=2
```