

HackerRank assessment Aspirant Handbook

HackerRank Assessment

To test the knowledge and skill acquired thru the learning, a HackerRank based assessment will be conducted. The assessment contains Skill and Knowledge based questions, running for 2 hours. The skill based questions contains questions that can be completed on the HackerRank platform provided code editor. Knowledge based questions are Multiple choice questions. Listed below are the topics.

Java track

1. Data structures and Algorithms
2. Core Java
3. Database

Multiple choice questions to test the knowledge on

1. WebUI – HTML, CSS, Javascript, Bootstrap, JQuery
2. Spring Core, Spring MVC, JDBC and JUnit

DotNet track

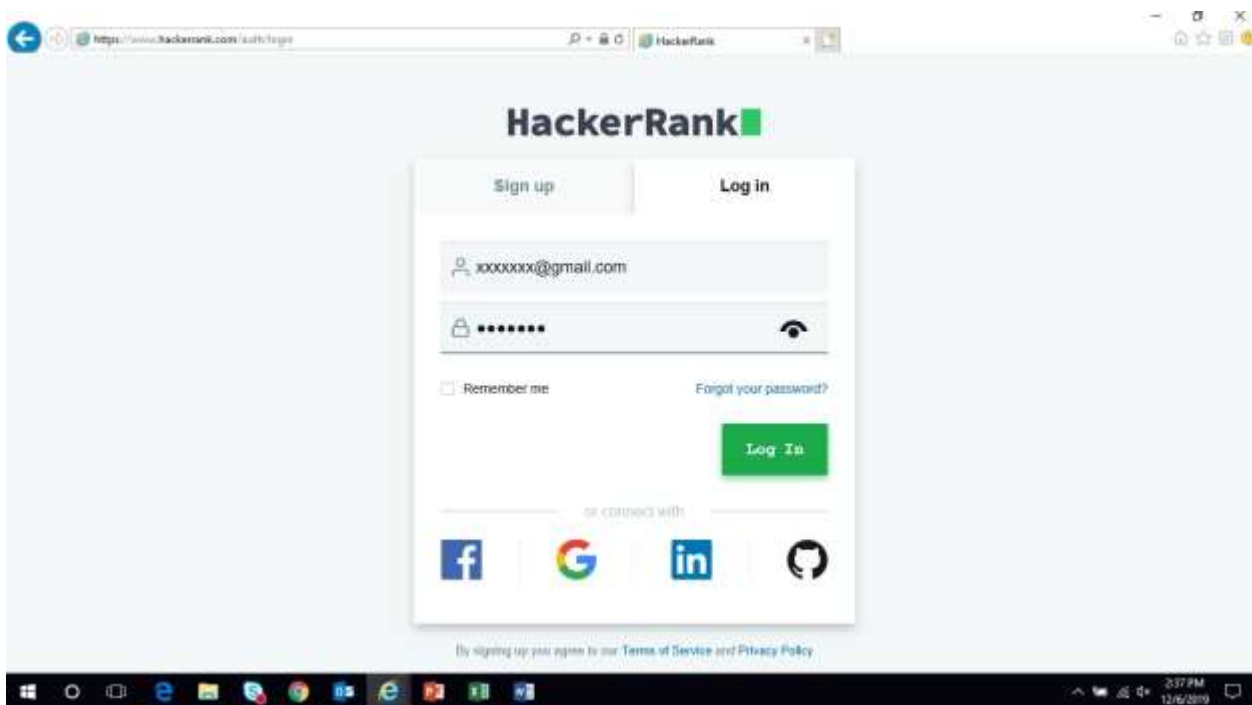
1. Data structures and Algorithms
2. C#
3. Database

Multiple choice questions to test the knowledge on

1. WebUI – HTML, CSS, Javascript, Bootstrap, JQuery
2. ASP.Net MVC, Entity Framework NUnit

Prep-up Login

1. Sign up through <https://www.hackerrank.com/signup>. If you already have an account, Log In. If you don't have an account, use your Personal Accounts (Gmail/Facebook) to sign up. Alternatively you can sign up by creating a new HackerRank Account.

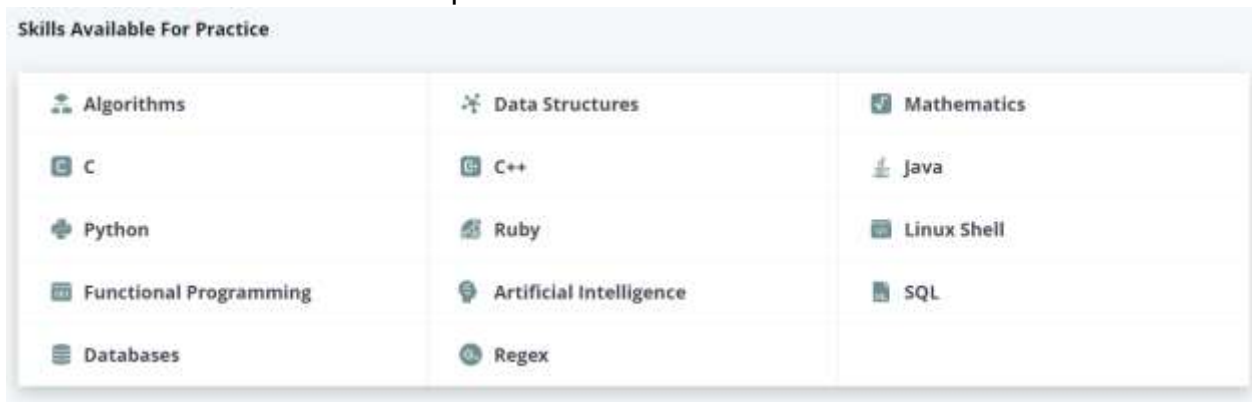


Prep-up on Java/C#, Problem solving and Data structures

1. Once you sign-in, you can explore the Skill as Java or Algorithms by navigating to the section shown below.

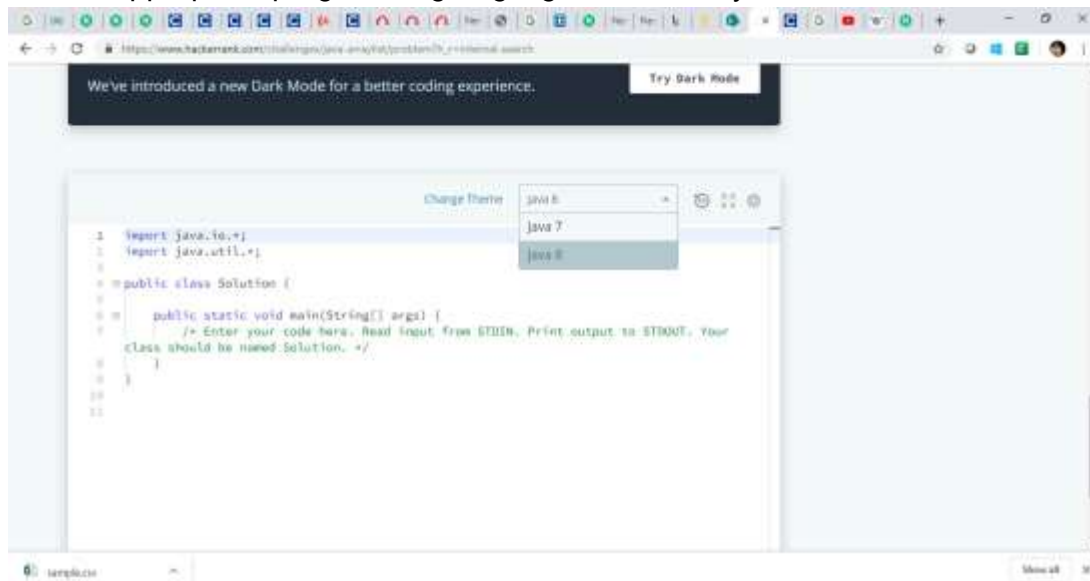
Any question on Algorithms can be solved thru Java or C#. This tests your proficiency in the Problem solving area.

Choose Data structures to choose problems on them.

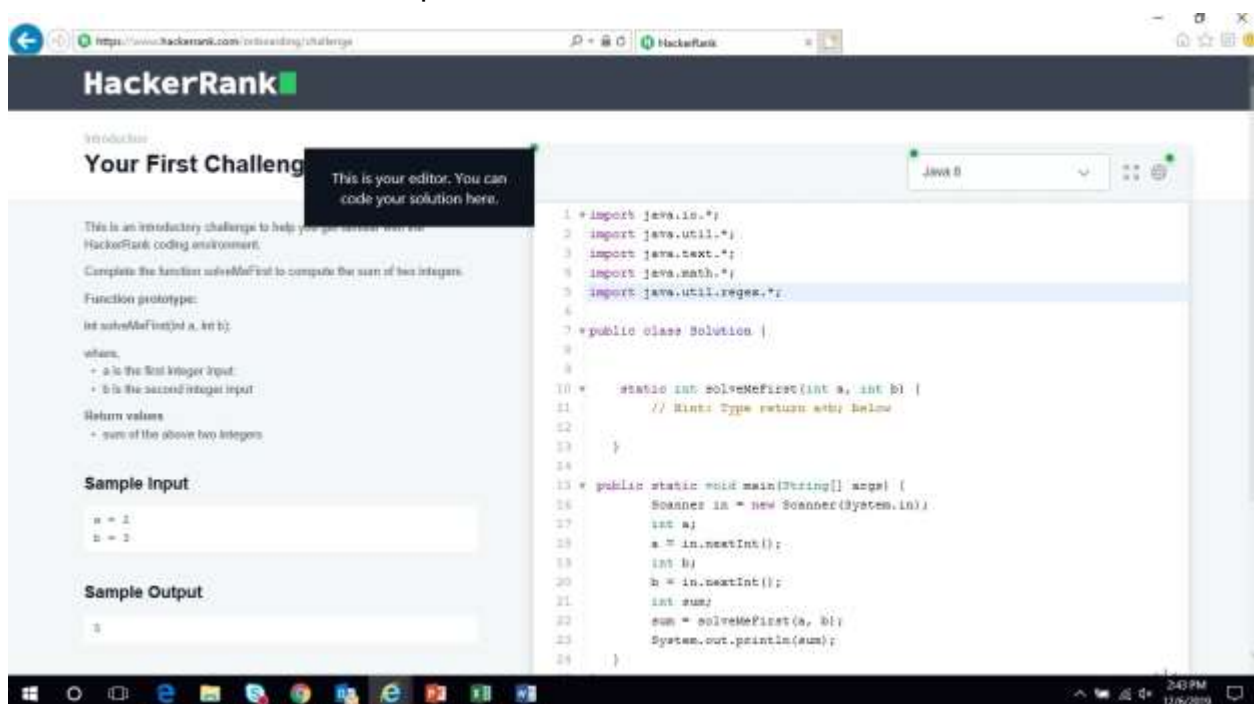


2. Start solving the problems using the Online Code editor and run your code and verify the results.

Choose the appropriate programming language filter and try out the code.



3. Use Code editor to solve the problem statement



4. You can try running the code using “Run Code” button

The screenshot shows a web browser window displaying the HackerRank challenge 'Solve Me First'. The page includes a problem description, a function prototype, sample input/output, and an explanation. A code editor is open, showing a Java solution. The code defines a static method `solveMeFirst` that takes two integers `a` and `b` and returns their sum. The `main` method uses a `Scanner` to read input from `System.in`, calls `solveMeFirst`, and prints the result. A green 'Run Code' button is visible at the bottom right of the code editor.

Complete the function `solveMeFirst` to compute the sum of two integers.

Function prototype:

```
int solveMeFirst(int a, int b);
```

where,

- `a` is the first integer input.
- `b` is the second integer input.

Return values

- sum of the above two integers

Sample Input

```
a = 2
b = 3
```

Sample Output

```
5
```

Explanation

The sum of the two integers `a` and `b` is computed as: $2 + 3 = 5$.

```
1 import java.text.*;
2 import java.math.*;
3 import java.util.regex.*;
4
5 public class Solution {
6
7     // Complete the solveMeFirst function below.
8
9     // Hint: Type return a+b; below
10
11     static int solveMeFirst(int a, int b) {
12
13     }
14
15     public static void main(String[] args) {
16         Scanner in = new Scanner(System.in);
17         int a;
18         a = in.nextInt();
19         int b;
20         b = in.nextInt();
21         int sum;
22         sum = solveMeFirst(a, b);
23         System.out.println(sum);
24     }
25 }
```

Line: 15 Col: 42

Run Code

Facing issues? Contact us [here](#) for support.

The code upon running will show the test cases that are associated, the expected and actual result. This can be referenced to check the logic and fix any bug.

Sample Test

You can take up this [sample test](#), any time after you finish the Practice.