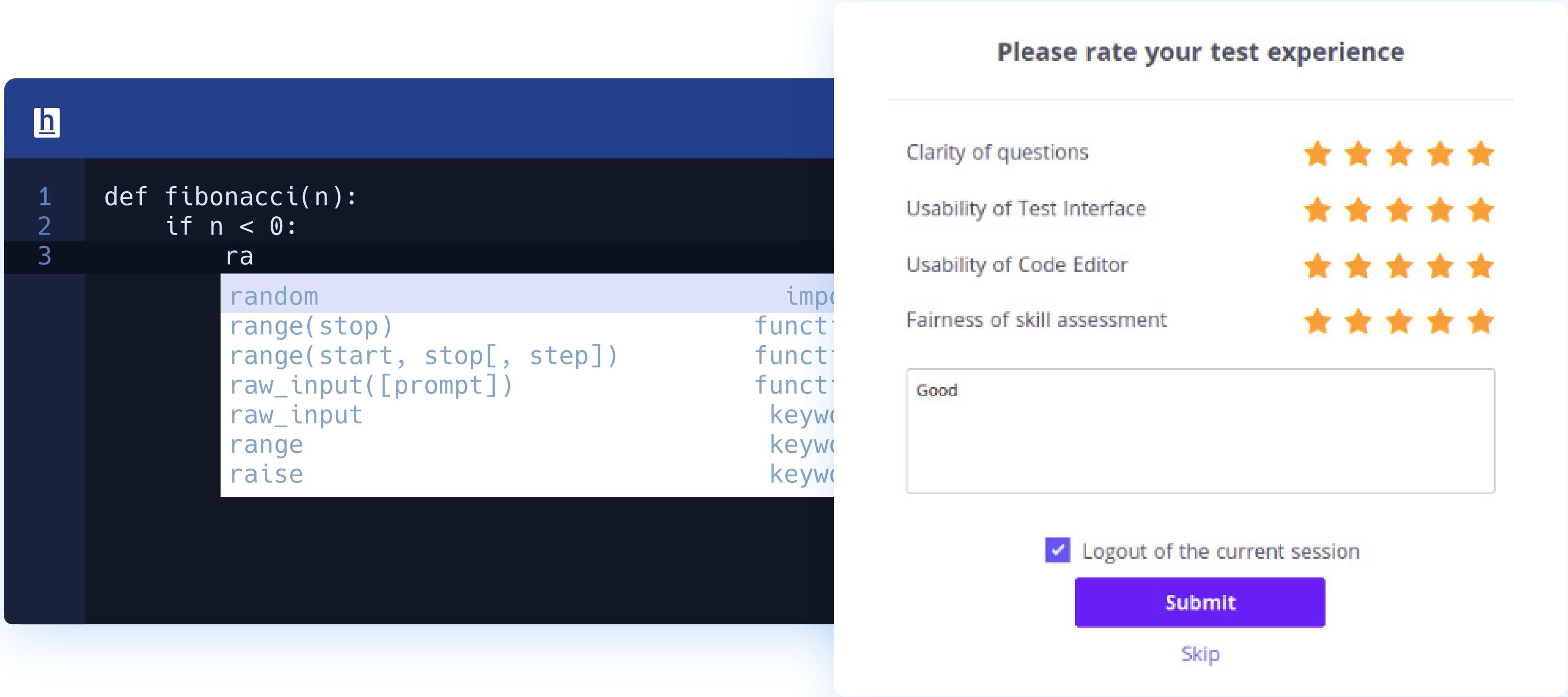
hackerearth



```
def fibonacci(n):
    if n < 0:
        ra
random                               impo
range(stop)                        funct
range(start, stop[, step])         funct
raw_input([prompt])                funct
raw_input                           keywo
range                               keywo
raise                               keywo
```

Please rate your test experience

Clarity of questions	★★★★★
Usability of Test Interface	★★★★★
Usability of Code Editor	★★★★★
Fairness of skill assessment	★★★★★

Good

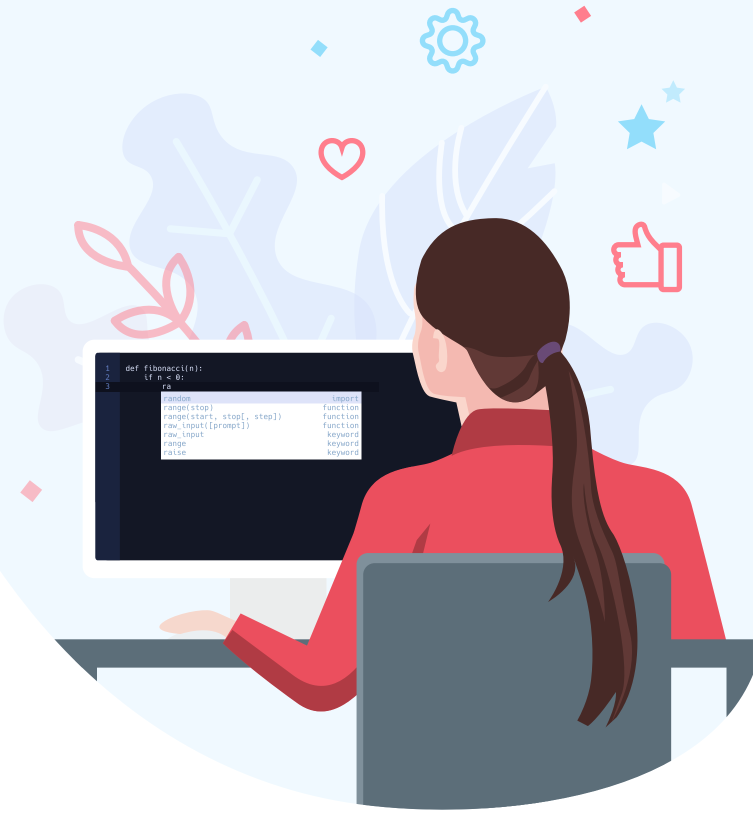
☑ Logout of the current session

How to improve **candidate experience** using HackerEarth

re

Enhancing candidate experience using developer assessments

When an organization evaluates an assessment platform for their screening process, they also look at how optimal the candidate experience is going to be. A negative candidate experience can not only lead to organizations missing out on hiring potential candidates, but also incur a significant loss of trust and revenue. On the contrary, a positive experience can benefit the organization in building a strong talent pipeline.



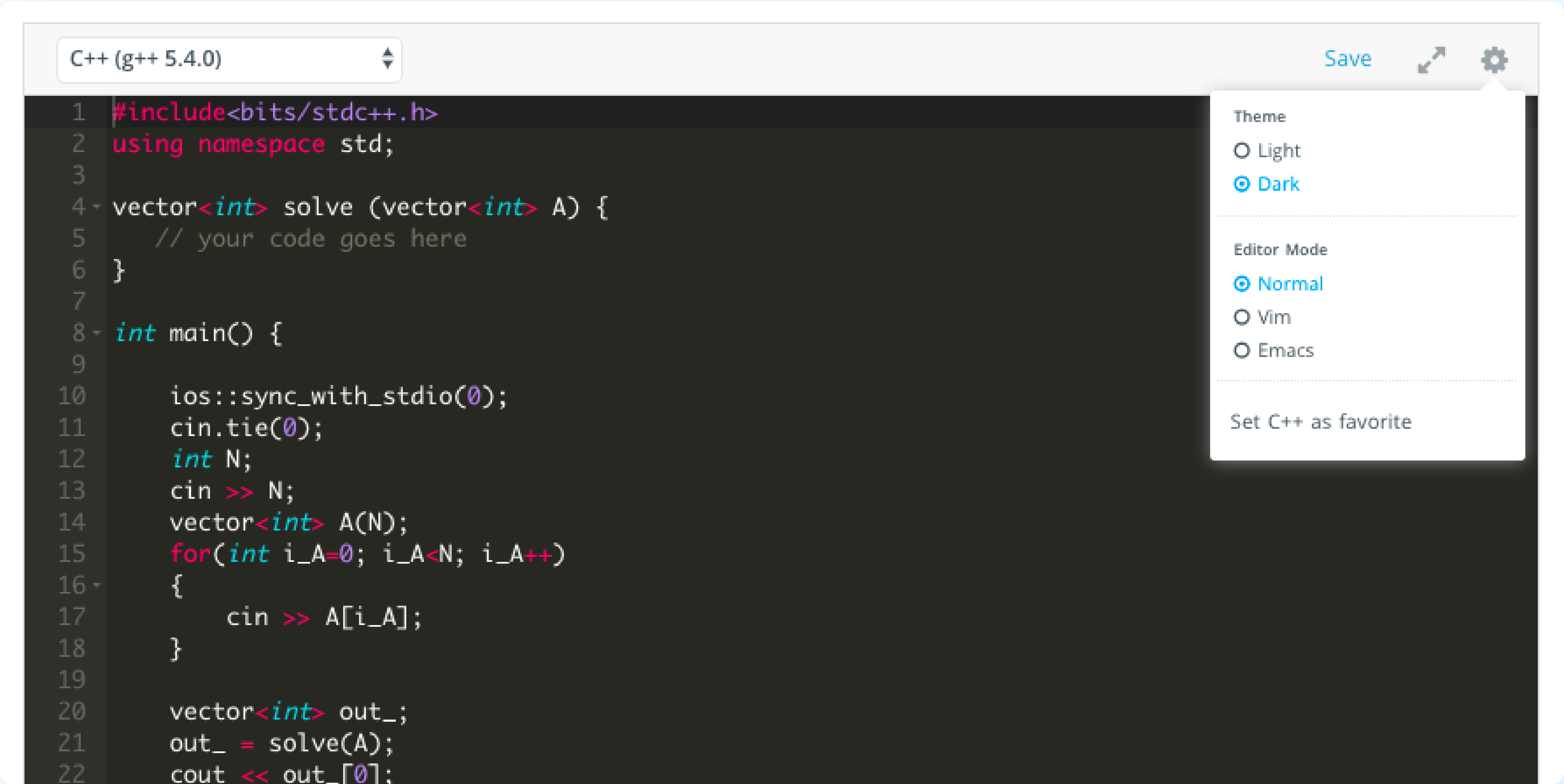
```
def fibonacci(n):
    if n < 0:
        ra
        random                          import
        range(stop)                     function
        range(start, stop[, step])      function
        raw_input([prompt])             function
        raw_input                        keyword
        range                            keyword
        raise                            keyword
```

Ensuring a good candidate experience is a crucial part of an organization's talent acquisition strategy. Thus, they need to be watchful of the assessment platform they use in their hiring process. Here's how you can use HackerEarth to provide an ideal test experience for candidates.

Bring out the best in your candidates

Allow candidates to code more efficiently using online IDEs

The HackerEarth platform offers multiple IDE options (such as VIM and Emacs) that are available for a smooth user interface. These options enable candidates to write,debug, and compile their code more effectively. Also, based on their preference, candidates can choose to toggle between editor modes, code in a light or dark theme, and even set a programming language as their favorite.



```cpp
#include<bits/stdc++.h>
using namespace std;

vector<int> solve (vector<int> A) {
    // your code goes here
}

int main() {

    ios::sync_with_stdio(0);
    cin.tie(0);
    int N;
    cin >> N;
    vector<int> A(N);
    for(int i_A=0; i_A<N; i_A++)
    {
        cin >> A[i_A];
    }

    vector<int> out_;
    out_ = solve(A);
    cout << out_[0];
```

Theme

○ Light
● Dark

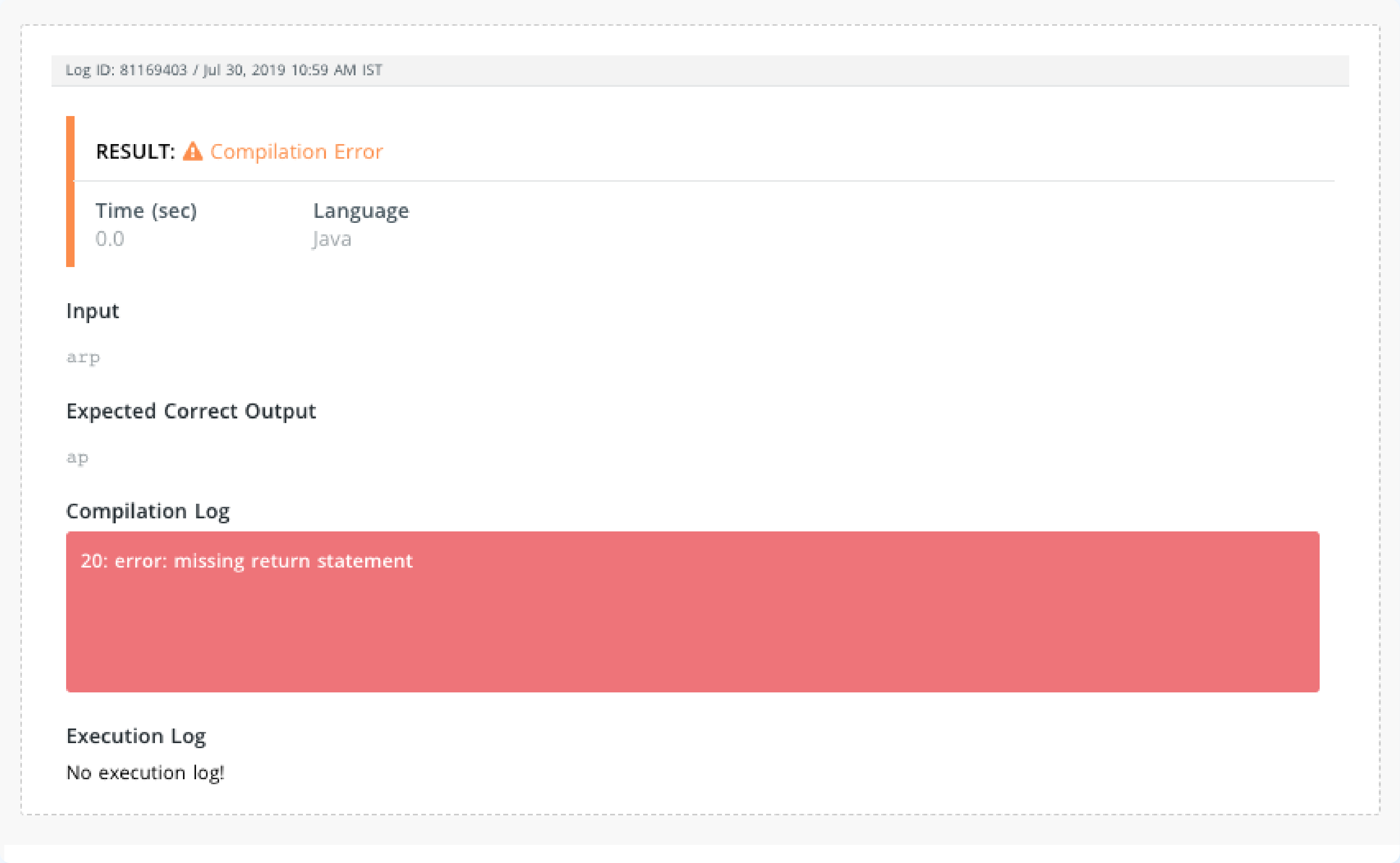
Editor Mode

● Normal
○ Vim
○ Emacs

Set C++ as favorite

Save a candidate's time while coding

While executing their code, candidates can directly view compilation and runtime errors below the code editor interface, making it easier for candidates to fix their code before they submit it.



Log ID: 81169403 / Jul 30, 2019 10:59 AM IST

RESULT: ⚠ Compilation Error

Time (sec)
0.0

Language
Java

Input

```
arp
```

Expected Correct Output

```
ap
```

Compilation Log

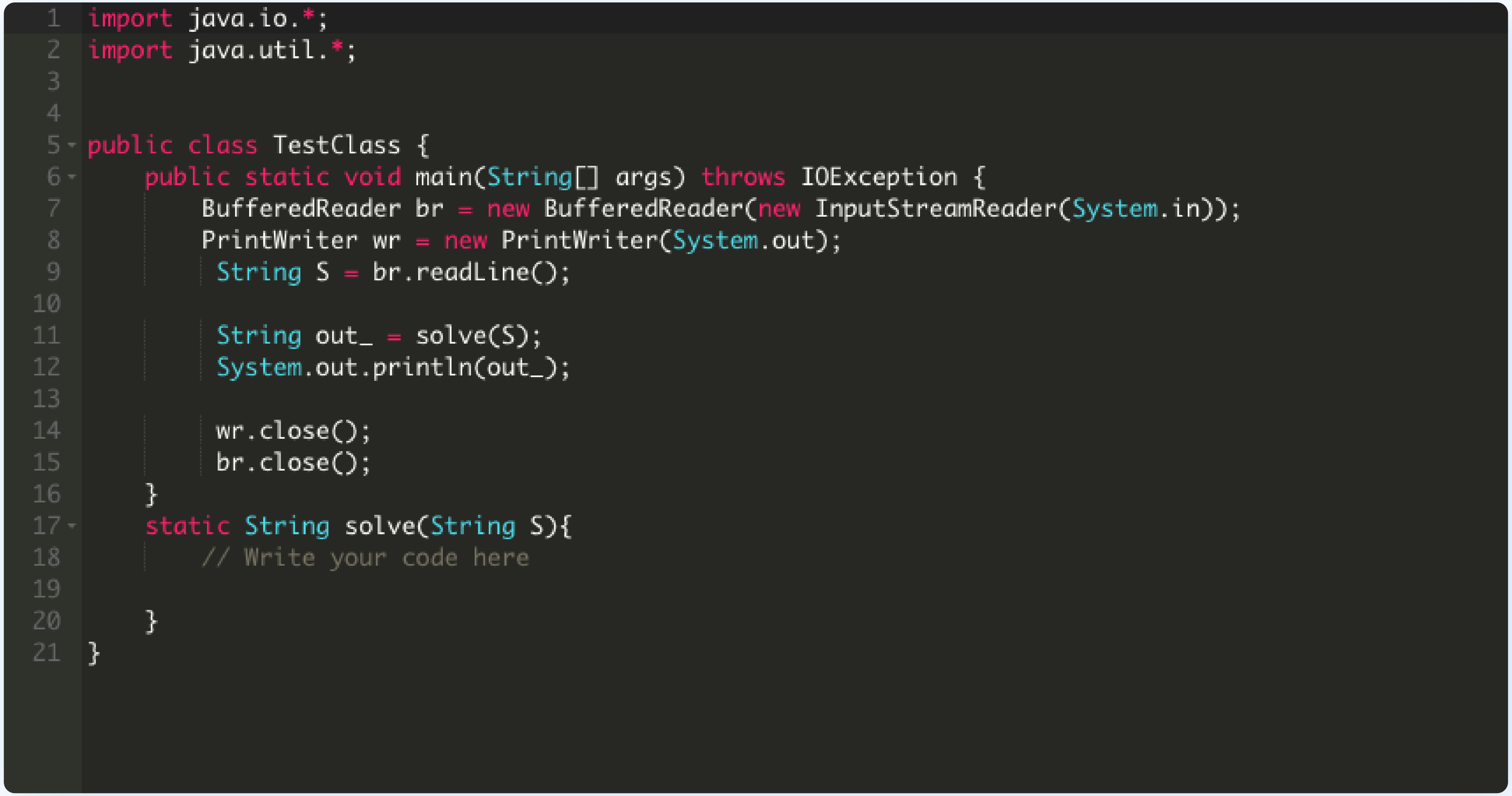
```
20: error: missing return statement
```

Execution Log

No execution log!

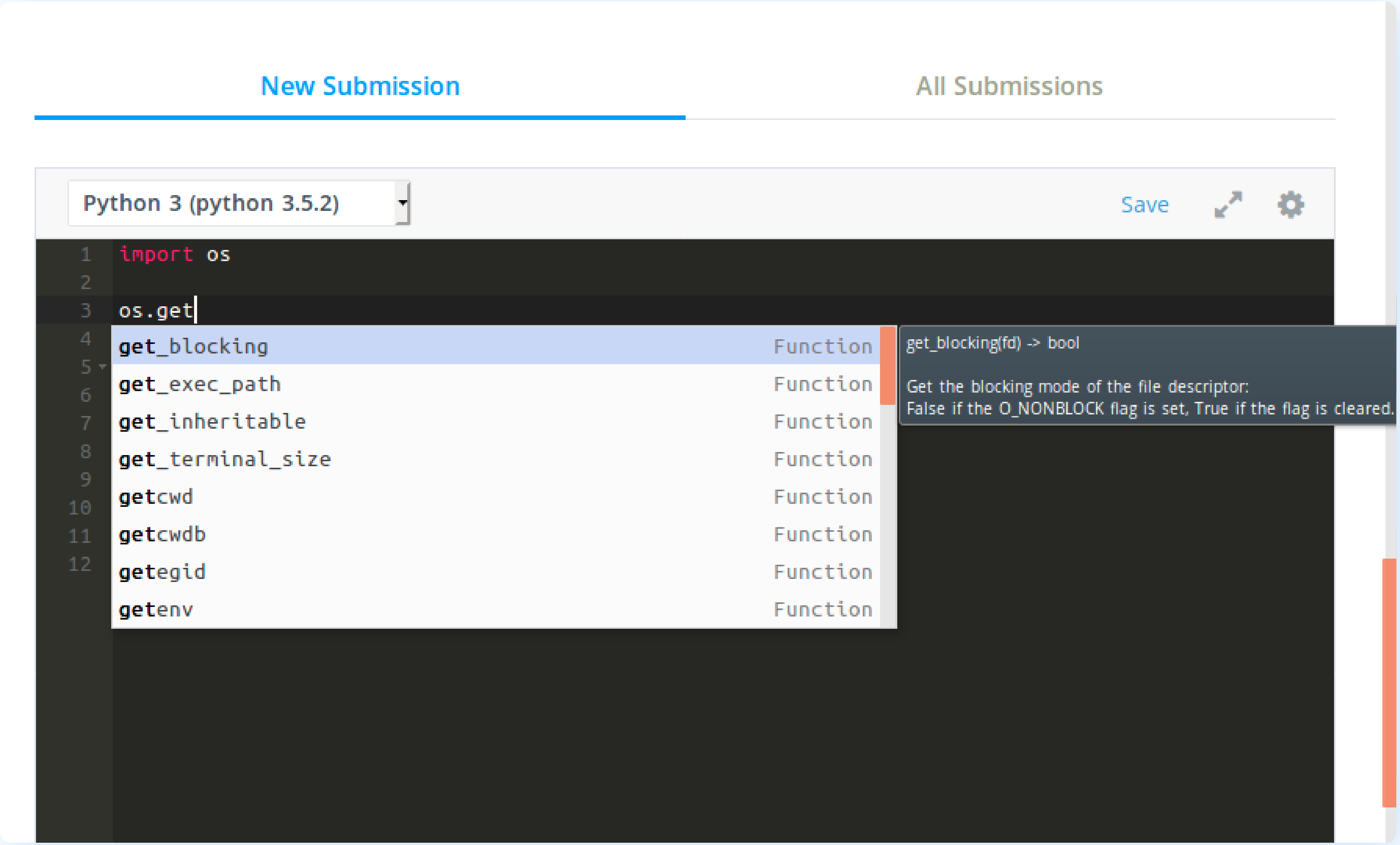
In addition, the pre-populated code snippets and autocomplete features in the HackerEarth platform let candidates focus on logic rather than syntax.

Sample code snippet for Java



```java
import java.io.*;
import java.util.*;


public class TestClass {
    public static void main(String[] args) throws IOException {
        BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
        PrintWriter wr = new PrintWriter(System.out);
         String S = br.readLine();

         String out_ = solve(S);
         System.out.println(out_);

         wr.close();
         br.close();
    }
    static String solve(String S){
        // Write your code here

    }
}
```

Using autocomplete feature in HackerEarth



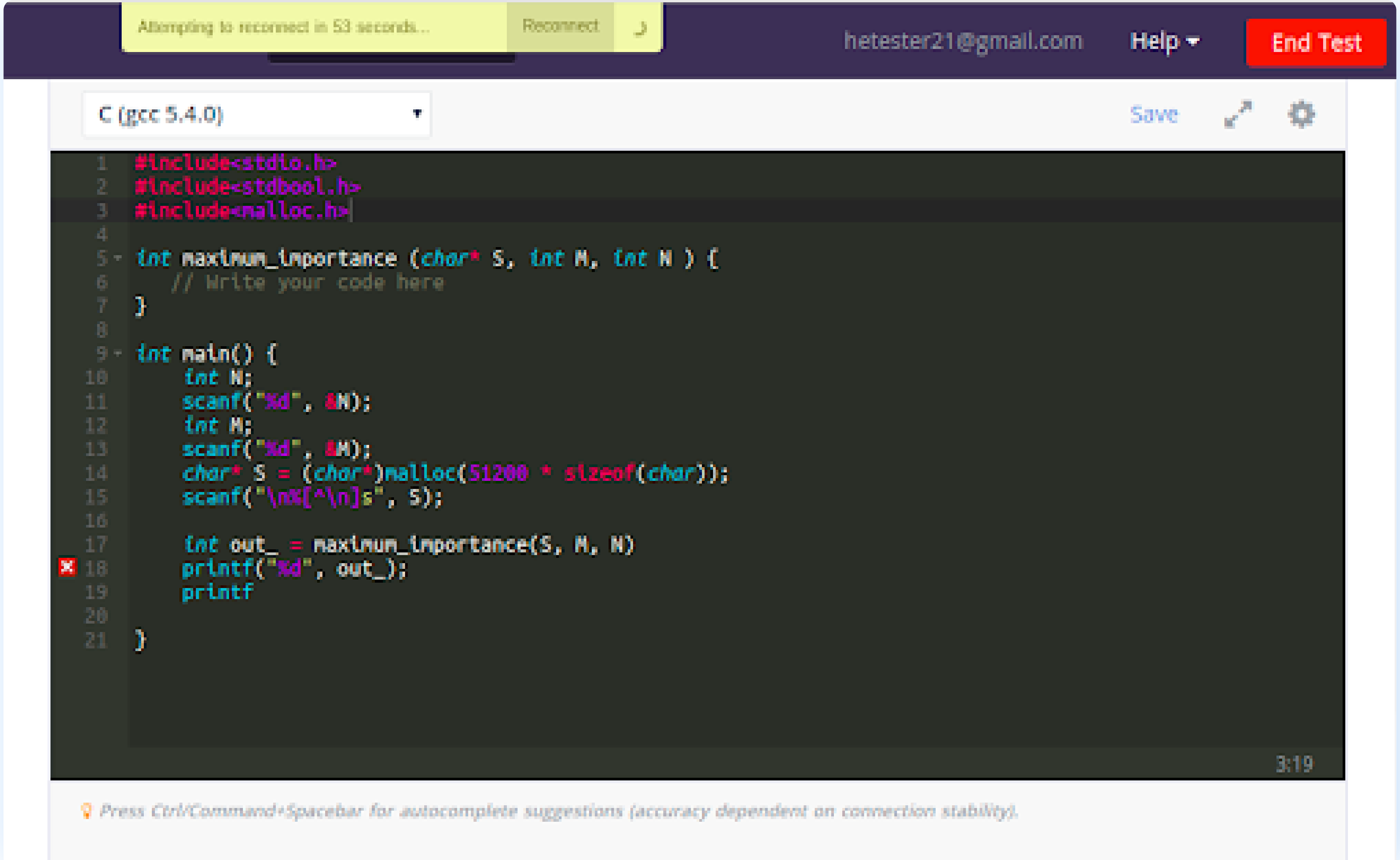
```
import os

os.get
```

get_blocking(fd) -> bool

Get the blocking mode of the file descriptor:
False if the O_NONBLOCK flag is set, True if the flag is cleared.

Provide a smooth test experience using proactive alerts

If there’s an error in the test environment, candidates receive alerts with appropriate error messages. These can include issues related to network failure, server error, errors in loading files, and more.



Attempting to reconnect in 53 seconds...

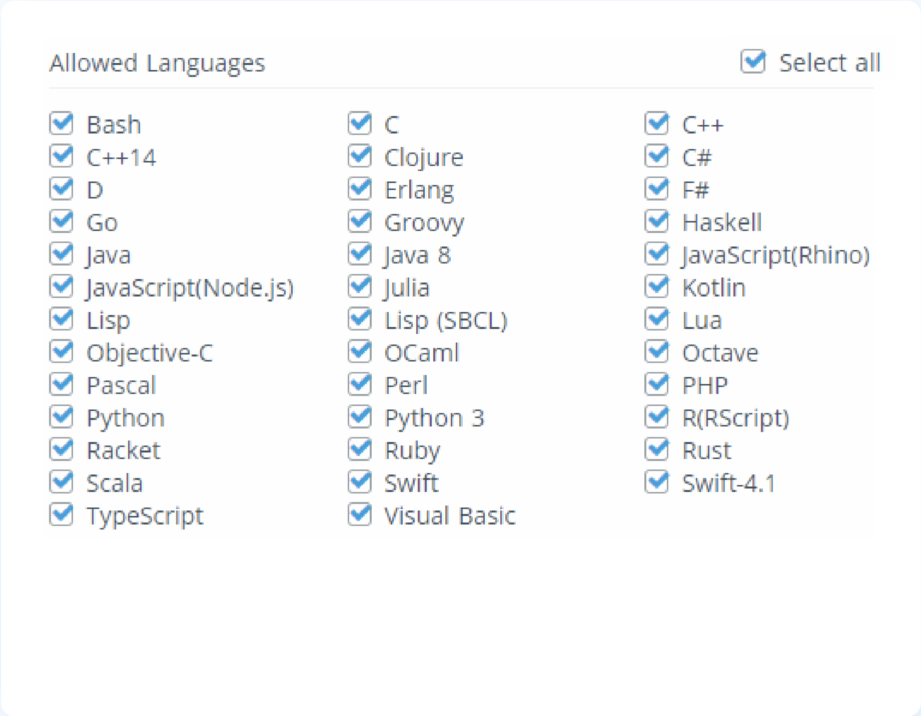
```
#include<stdio.h>
#include<stdbool.h>
#include<malloc.h>

int maximum_importance (char* S, int M, int N ) {
    // Write your code here
}

int main() {
    int N;
    scanf("%d", &N);
    int M;
    scanf("%d", &M);
    char* S = (char*)malloc(51200 * sizeof(char));
    scanf("\n%[^\n]s", S);

    int out_ = maximum_importance(S, M, N)
    printf("%d", out_);
    printf

}
```

Press Ctrl/Command+Spacebar for autocomplete suggestions (accuracy dependent on connection stability).

Make assessments more user-friendly

Code in the language of their choice

Based on admin settings, candidates can choose from 38 different programming languages to answer programming questions

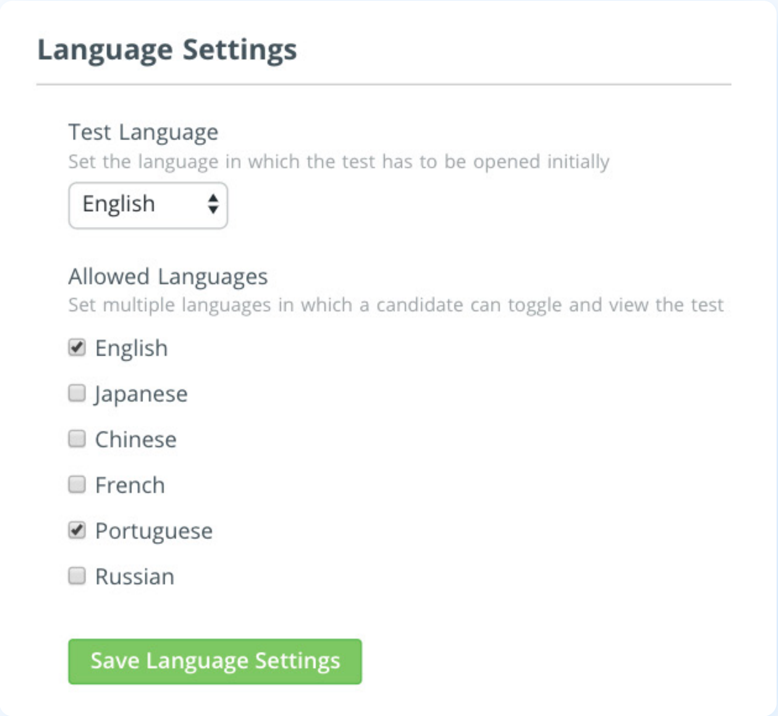


Allowed Languages ☑ Select all

- ☑ Bash
- ☑ C++14
- ☑ D
- ☑ Go
- ☑ Java
- ☑ JavaScript(Node.js)
- ☑ Lisp
- ☑ Objective-C
- ☑ Pascal
- ☑ Python
- ☑ Racket
- ☑ Scala
- ☑ TypeScript
- ☑ C
- ☑ Clojure
- ☑ Erlang
- ☑ Groovy
- ☑ Java 8
- ☑ Julia
- ☑ Lisp (SBCL)
- ☑ OCaml
- ☑ Perl
- ☑ Python 3
- ☑ Ruby
- ☑ Swift
- ☑ Visual Basic
- ☑ C++
- ☑ C#
- ☑ F#
- ☑ Haskell
- ☑ JavaScript(Rhino)
- ☑ Kotlin
- ☑ Lua
- ☑ Octave
- ☑ PHP
- ☑ R(RScript)
- ☑ Rust
- ☑ Swift-4.1

Take assessments in different languages

The HackerEarth platform allows candidates to take an assessment in five other native languages apart from English



Language Settings

Test Language

Set the language in which the test has to be opened initially

English

Allowed Languages

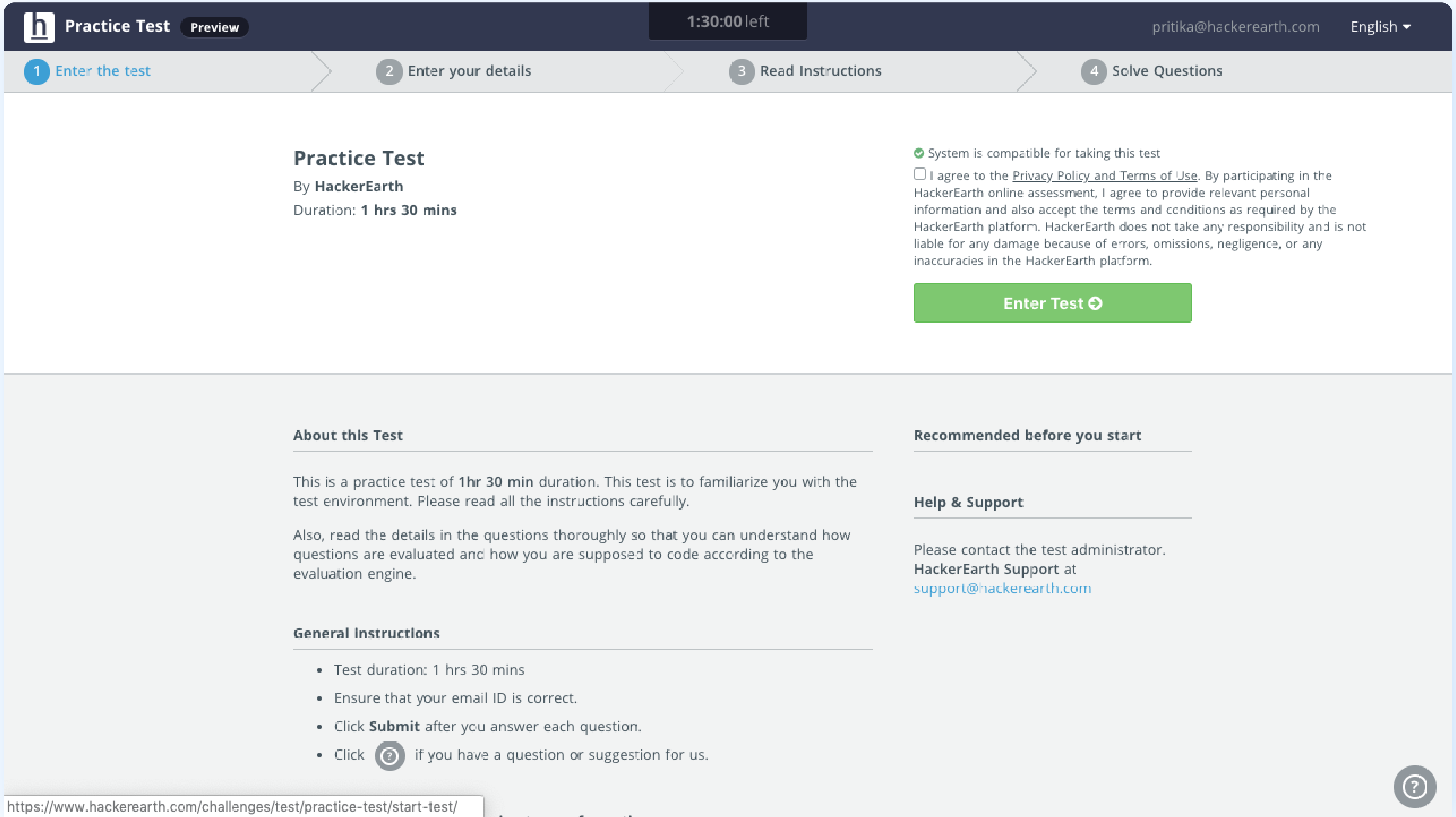
Set multiple languages in which a candidate can toggle and view the test

- ☑ English
- □ Japanese
- □ Chinese
- □ French
- ☑ Portuguese
- □ Russian

Provide room for improvement

Prepare for assessments by taking practice tests

Candidates can practice tests and get familiar with the test environment. This will help them in honing their skills before attempting an actual assessment.



Practice Test

By **HackerEarth**

Duration: **1 hrs 30 mins**

✓ System is compatible for taking this test

□ I agree to the Privacy Policy and Terms of Use. By participating in the HackerEarth online assessment, I agree to provide relevant personal information and also accept the terms and conditions as required by the HackerEarth platform. HackerEarth does not take any responsibility and is not liable for any damage because of errors, omissions, negligence, or any inaccuracies in the HackerEarth platform.

Enter Test

About this Test

This is a practice test of **1hr 30 min** duration. This test is to familiarize you with the test environment. Please read all the instructions carefully.

Also, read the details in the questions thoroughly so that you can understand how questions are evaluated and how you are supposed to code according to the evaluation engine.

General instructions

- Test duration: 1 hrs 30 mins
- Ensure that your email ID is correct.
- Click **Submit** after you answer each question.
- Click ? if you have a question or suggestion for us.

Recommended before you start

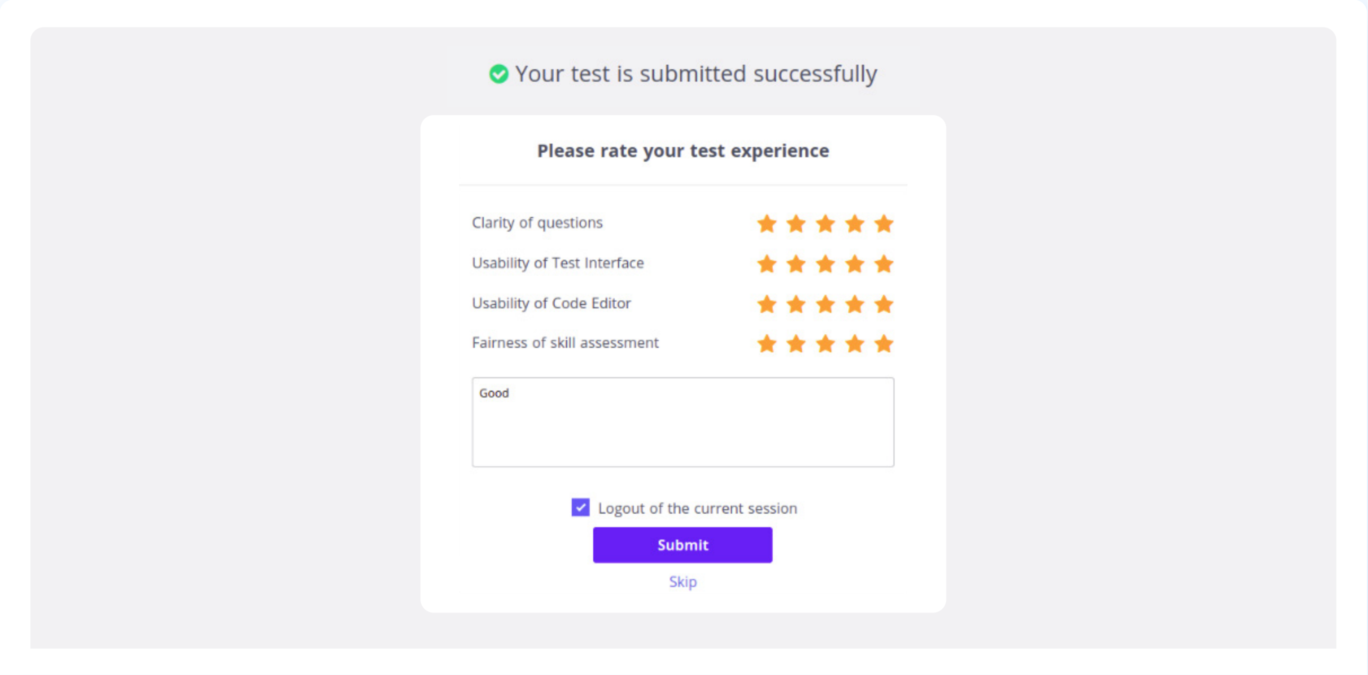
Help & Support

Please contact the test administrator.
HackerEarth Support at
support@hackerearth.com

https://www.hackerearth.com/challenges/test/practice-test/start-test/

Make your assessments more relevant with candidate feedback

After completing an assessment, candidates can give detailed feedback on the HackerEarth platform, informing recruiters about their experience and if there's any scope of improvement.



Your test is submitted successfully

Please rate your test experience

Clarity of questions	★★★★★
Usability of Test Interface	★★★★★
Usability of Code Editor	★★★★★
Fairness of skill assessment	★★★★★

Good

☑ Logout of the current session

Try taking an assessment on HackerEarth and let us know how it works for you. If you need any help using the HackerEarth platform, write back to us at **support@hackerearth.com**. If you're new to HackerEarth and want to create accurate skill-based developer assessments, **sign up for a 14-day free trial**.