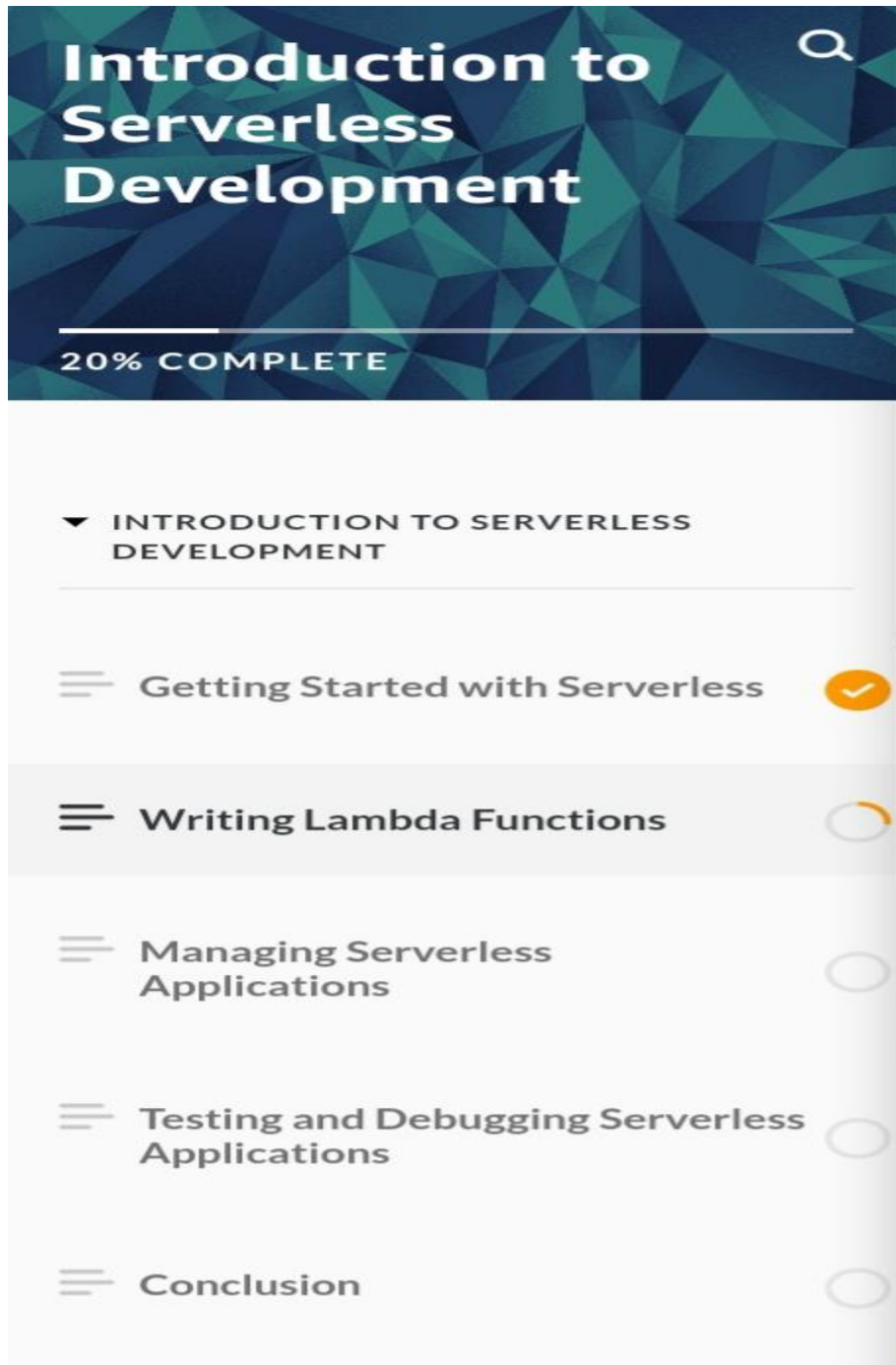



DAILY ONLINE ACTIVITIES SUMMARY

Date:	24/06/2020	Name:	Prathiksha
Sem & Sec	8 th sem & B sec	USN:	4AL16CS070
Online Test Summary			
Subject	-		
Max. Marks	-	Score	-
Certification Course Summary			
Course	Introduction to serverless development.		
Certificate Provider	AWS	Duration	25 min
Coding Challenges			
Problem Statement: 1. Program to Sort a stack using a temporary stack in java.			
Status: Solved			
Uploaded the report in Github		Yes	
If yes Repository name		Prathiksha	
Uploaded the report in slack		Yes	

Online Test Details:






Certification Course Details:

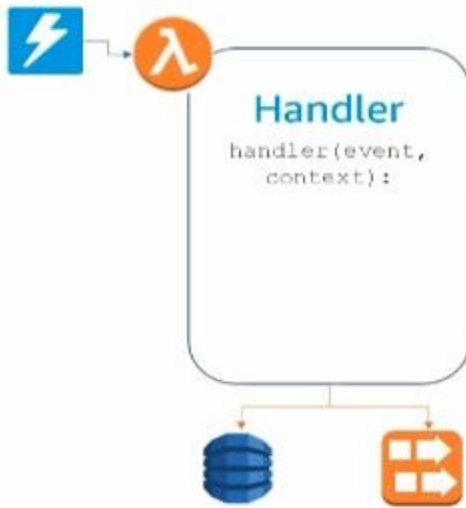


Introduction to Serverless Development 

20% COMPLETE

▼ INTRODUCTION TO SERVERLESS DEVELOPMENT

- ≡ Getting Started with Serverless 
- ≡ Writing Lambda Functions 
- ≡ Managing Serverless Applications 
- ≡ Testing and Debugging Serverless Applications 
- ≡ Conclusion 



© 2018 Amazon Web Services, Inc. or its Affiliates. All rights reserved.

Every Lambda function has a Handler method

- Handler is like **Java MAIN**
- **Limit** what you include

Key Developer Considerations

Writing Code

- System architecture
- Design patterns
- Frameworks and libraries

Managing Code

- Tools (IDEs, SCM, debuggers, etc.)
- Developer workflow
- Test/deployment automation
- Environment management

© 2018 Amazon Web Services, Inc. or its Affiliates. All rights reserved.

Topic: About serverless development.

Coding Challenges Details:

Program 1:

```
import java.util.*;

class SortStack
{
    public static Stack<Integer> sortstack(Stack<Integer>
                                           input)
    {
        Stack<Integer> tmpStack = new Stack<Integer>();
        while(!input.isEmpty())
        {
            int tmp = input.pop();

            while(!tmpStack.isEmpty() && tmpStack.peek()
                  > tmp)
            {

                input.push(tmpStack.pop());
            }

            // push temp in tempory of stack
            tmpStack.push(tmp);
        }
        return tmpStack;
    }

    public static void main(String args[])
    {
        Stack<Integer> input = new Stack<Integer>();
        input.add(34);
        input.add(3);
        input.add(31);
        input.add(98);
        input.add(92);
        input.add(23);

        // This is the temporary stack
        Stack<Integer> tmpStack=sortstack(input);
        System.out.println("Sorted numbers are:");
    }
}
```

```
        while (!tmpStack.empty())
        {
            System.out.print(tmpStack.pop()+" ");
        }
    }
}
```

Out put:

Sorted numbers are:

3 23 31 34 92 98