# CS F213 : OOP Lab 5
# Abstract Classes

Shubham Agrawal, Arapan Baranwal, Manas Mhasakar

23rd March 2021

In this lab we will be focusing on the concept of abstract classes. The objective of this lab will be to simulate a cooking competition and predict the winner.

**General Instructions**

Read the following instructions very carefully before starting your work -

1. Make sure that the source code is written in the default package.
2. Ensure that the name of each file is the same as the corresponding class name.
3. Ensure that the name of methods is the same as those given in Javadoc. Failure to comply may cause deduction in credit even if the implementation is correct.
4. You won't be provided with a template file (helper code). You have to implement all the classes by yourself (including their respective constructors).
5. Do not copy directly the names of methods from Javadoc. Write them out yourselves.
6. You are required to upload all the classes at once. It is not possible to test individual classes on Codepost.
7. A brief description of the classes and methods to be used is given further in the problem statement but you must refer to the Javadoc for the complete description.

8. Be sure to check your code on the test cases provided locally (as jar file), before submitting on Codepost. Remember that Codepost takes

only the latest submission and not the best submission into consideration.

9. The advisable order of writing the code:

a) Cook
b) Dish
c) Chef
d) Apprentice
e) MainCourse
f) Starter
g) Team
h) Competition

A brief description of the classes involved are mentioned below.

1) Two abstract classes named ***Cook*** and ***Dish***.
2) Each Cook can either be a ***Chef*** or an ***Apprentice***.
3) Each dish can either be a ***Starter*** or a ***MainCourse.***
4) A ***Team*** consists of a Chef, Apprentice and a Dish.
5) A ***Competition*** consists of a collection of different Teams***.***

You will have to implement various methods in each of these classes as per the javadoc in order to successfully simulate the competition.

You have to submit all eight classes on Codepost.

# Class Descriptions

## 1.Competition

- ***Field Summary***
  - MAX_TEAMS
  - teams  (An array of Teams)
  - numTeams
  - competitionType
- ***Constructor***
  - Competition(competitionType)
- ***Methods***
  - addTeam(team)
  - findWinner()
  - getNumTeams()
  - removeTeam(team)

## 2. Apprentice

- ***Field Summary***
  - experience
  - MIN_AGE
  - MIN_EXPERIENCE
- ***Constructor***
  - Apprentice(name, age, ID, experience)
- ***Methods***
  - calculateSkill()
  - getExperience()
  - verifyValidity()

3. **Chef**

- ***Field Summary***
  - MAX_COMPETITIONS
  - MIN_AGE
  - MIN_POINTS
  - numCompetitions
  - points
- ***Constructor***
  - Chef(name, age, ID, points, numCompetitions)
- ***Methods***
  - calculateSkill()
  - getNumCompetitions()
  - getPoints()
  - participateInNewComp()
  - removeFromComp()
  - verifyValidity()

4. **Cook**

- ***Field Summary***
  - age
  - ID
  - name
  - skill
- ***Constructor***
  - Cook(name, age, ID)
- ***Methods***
  - calculateSkill()
  - getSkill()
  - verifyValidity()

5. **Dish**

- ***Field Summary***
  - costOfIngredients
  - numIngredients
  - name
  - timeForPrep
- ***Constructor***
  - Dish(name, costOfIngredients, timeForPrep)
- ***Methods***
  - verifyValidity()

## 6. MainCourse

- ***Field Summary***
  - MAIN_COURSE_MAX_COST
  - MAIN_COURSE_MAX_TIME
- ***Constructor***
  - MainCourse(name, costOfIngredients, timeForPrep)
- ***Methods***
  - verifyValidity()

## 7. Starter

- ***Field Summary***
  - STARTER_MAX_COST
  - STARTER_MAX_TIME
- ***Constructor***
  - Starter(name, costOfIngredients, timeForPrep)
- ***Methods***
  - verifyValidity()

## 8. Team

- ***Field Summary***
  - apprentice
  - dish
  - chef
  - teamSkill
- ***Constructor***
  - Team(chef, apprentice, mainCourse)
  - Team(chef, apprentice, starter)
- ***Methods***
  - calculateTeamSkill()
  - getChef()
  - getTeamSkill()
  - verifyValidity(competitionType)

Marks Distribution for the TEST CASES

| Methods | Marks Allotted |
|---|---|
| verifyValidityApprentice | 0.5 |
| verifyValidityChef | 0.5 |
| calculateSkillApprentice<br>calculateSkillChef | 0.5 |
| verifyValidityDish | 0.5 |
| verifyValidityTeam | 0.5 |
| calculateTeamSkill | 0.5 |
| AddTeam | 1 |
| RemoveTeam | 1 |
| FindWinner | 2 |
| Total | 7 |