

Scala Peer Learning

Rajkeshav Kumar Jha

TAS129

Q1)

Anuj's Approach

- `get_bucket_range` function is used to find the bucket for a respective number.
- It gets the number behind the decimal point and checks if it is greater than equal to 50 or less than 50.
- Accordingly it finds the bucket range for the number by finding its left and right endpoints.

Pankaj's Approach

- Created `get_bucket()` function to find the bucket in which the given decimal number falls.
- Found the last two digits of the decimal number then Checked if it is less than 50 or greater than or equal to 50.
- If it is lesser than 50 , it will fall into bucket category of 00 - 49 $[n - (\text{last_two_digits} / 1000) \text{ to } n + (49 - \text{last_two_digits}) / 1000]$
- Otherwise it will fall into the category of 50 - 99 $[n - (\text{last_two_digits} / 1000) + 0.050 \text{ to } n + (99 - \text{last_two_digits}) / 1000]$.

Q2)

Anuj's Approach

- Created a trait for the player which contains all its attributes.
- Created a case class which extends the trait, which gives the advantage of not creating an instance for the class.

- Function `get_player_info` gives information about players.
- Used `Source` object to read input data from the file and converted it into a list.
- Added players as the `Players` object and then performed queries on it.

Pankaj's Approach

- defined a case class `Player` with fields for year, name, country, matches, runs, and wickets. then created a sample sequence of `Player` instances for demonstration purposes.
- then proceeds to perform several operations on the players List:
 - The player with the highest run scored is found by calling the `maxBy` method on the players List with the runs field as the key.
 - The top 5 players by run scored are found by calling the `sortBy` method on the players with the runs field as the key, in descending order and printing the first 5 elements using `While` loop.
 - The top 5 players by wickets are found by calling the `sortBy` method on the players sequence with the wickets field as the key, in descending order and then printing the first 5 elements using the `While()` loop. .
 - Ranked the players by overall performance, defined as $\text{runs} + (\text{wickets} * 5)$ by mapping each player to a List of the player instance and their overall performance score, then sorting the resulting List of lists by the performance score in descending order. then printing out the rank, name, number of runs scored, number of wickets taken, and overall performance score for each player.