Class: BadCyclingPortalImpl
- teams: List<Team>
- races: List<Race>
- stages: List<Stage>
- riders: List<Rider>
+ getRaceIds(): int[]
+ createRace(name: String, description: String): int
+ viewRaceDetails(raceId: int): String
+ removeRaceById(raceId: int): void
+ getNumberOfStages(raceId: int): int
+ addStageToRace(raceId: int, stageName: String, description: String, length: double, startTime: LocalDateTime, type: StageType): int
+ getRaceStages(raceId: int): int[]
+ getStageLength(stageId: int): double
+ removeStageById(stageId: int): void
+ addCategorizedClimbToStage(stageId: int, location: Double, type: CheckpointType, averageGradient: Double, length: Double): int
+ addIntermediateSprintToStage(stageId: int, location: double): int
+ removeCheckpoint(checkpointId: int): void
+ concludeStagePreparation(stageId: int): void
+ getStageCheckpoints(stageId: int): int[]
+ createTeam(name: String, description: String): int
+ removeTeam(teamId: int): void
+ getTeams(): int[]
+ getTeamRiders(teamId: int): int[]
+ createRider(teamID: int, name: String, yearOfBirth: int): int
+ removeRider(riderId: int): void
+ registerRiderResultsInStage(stageId: int, riderId: int, checkpoints: LocalTime...): void
+ getRiderResultsInStage(stageId: int, riderId: int): LocalTime[]
+ getRiderAdjustedElapsedTimeInStage(stageId: int, riderId: int): LocalTime
+ deleteRiderResultsInStage(stageId: int, riderId: int): void
+ getRidersRankInStage(stageId: int): int[]
+ getRankedAdjustedElapsedTimesInStage(stageId: int): LocalTime[]
+ getRidersPointsInStage(stageId: int): int[]
+ getRidersMountainPointsInStage(stageId: int): int[]
+ eraseCyclingPortal(races: List<Race>, teams: List<Team>, stages: List<Stage>, riders: List<Rider>): void
+ saveCyclingPortal(filename: String): void
+ loadCyclingPortal(filename: String): void
+ removeRaceByName(name: String): void
+ getGeneralClassificationTimesInRace(raceId: int): LocalTime[]
+ getRidersPointsInRace(raceId: int): int[]
+ getRidersMountainPointsInRace(raceId: int): int[]
+ getRidersGeneralClassificationRank(raceId: int): int[]
+ getRidersPointClassificationRank(raceId: int): int[]
+ getRidersMountainPointClassificationRank(raceId: int): int[]
Races In Cycling Portal
Class: Race
- name: String
- description: String
- idCounter: int
- id: int
- allRaceIds: List<Integer>
+ races: List<Race>
+ stages: List<Stage>
+ Race(name: String, description: String)
+ getId(): int
+ getName(): String
+ getDescription(): String
+ getAllRaceIds(): List<Integer>
+ removeRaceById(raceId: int): void
+ addStage(stage: Stage): void
+ getNumberOfStages(raceId: int): int
+ getRaceStages(raceId: int): int[]
+ getStageLength(stageId: int): double
+ removeStageById(raceId: int): void
+ viewRaceDetails(raceId: int): String
+ concludeStagePreparation(stageId: int): void
+ getStageStartTime(stageId: int): LocalDateTime
+ getStageById(stageId: int): Stage
Stages in Race
Class: Stages
- state: String
- idCounter: int
- id: int
- raceId: int
- name: String
- description: String
- length: double
- startTime: LocalDateTime
- type: StageType
- checkpoints: List<Checkpoint>
+ Stage(raceId: int, name: String, description: String, length: double, startTime: LocalDateTime, type: StageType)
+ getId(): int
+ getRaceId(): int
+ getName(): String
+ getDescription(): String
+ getLength(): double
+ getStartTime(): LocalDateTime
+ getType(): StageType
+ getState(): String
+ setState(state: String): void
+ addCheckpoint(stageId: int, checkpoint: Checkpoint): void
+ removeCheckpoint(checkpointId: int): void
+ getStageCheckpoints(): int[]
+ getStageCheckpoints(stageId: int): int[]
Class: Stage State
- UNDER_PREPARATION
- WAITING_FOR_RESULTS
- CONCLUDED
Class: Stage Type
- FLAT
- MEDIUM_MOUNTAIN
- HIGH_MOUNTAIN
- TT
Checkpoints in stages
Class: Checkpoint
- idCounter: int
- id: int
- stageId: int
- location: double
- type: CheckpointType
- averageGradient: double
- length: double
+ addCategorizedClimbToStage(stageId: int, location: double, type: CheckpointType, averageGradient: double, length: double): int
+ addIntermediateSprintToStage(stageId: int, location: double): int
+ getId(): int
+ getStageId(): int
+ getLocation(): double
+ getType(): CheckpointType
+ getAverageGradient(): double
+ getLength(): double
+ getIdCounter(): int
Class: Checkpoint Type
- SPRINT
- C4
- C3
- C2
- C1
- HC
Teams In Race
Class: Teams
- name: String
- description: String
- nextTeamId: int
- id: int
- riders: List<Rider>
+ teams: List<Team>
+ Team(name: String, description: String)
+ getName(): String
+ getId(): int
+ addRider(name: String, yearOfBirth: int): int
+ removeRider(rider: Rider): void
+ getRiders(): List<Rider>
+ createTeam(name: String, description: String): int
+ removeTeam(teamId: int): void
+ getTeams(): int[]
+ getTeamRiders(teamId: int): int[]
Riders In Teams
Class: Riders
- teamID: int
- name: String
- yearOfBirth: int
- riderId: int
- nextRiderId: int
- resultsByStage: Map<Integer, LocalTime[]>
- riders: List<Rider>
+ Rider(teamID: int, name: String, yearOfBirth: int)
+ getNextRiderId(): int
+ getTeamID(): int
+ getId(): int
+ createRider(teamID: int, name: String, yearOfBirth: int): int
+ removeRider(riderId: int): void
+ setResultsForStage(stageId: int, checkpoints: LocalTime[]): void
+ hasResultsForStage(stageId: int): boolean
+ getResultsForStage(stageId: int): LocalTime[]
+ retrieveRiderById(riderId: int): Rider
+ removeResultsForStage(stageId: int): void
+ getAllParticipatingRiderIds(stageId: int): List<Integer>
+ sortRiderIdsByFinishTime(allRiderIds: List<Integer>, stageId: int): void