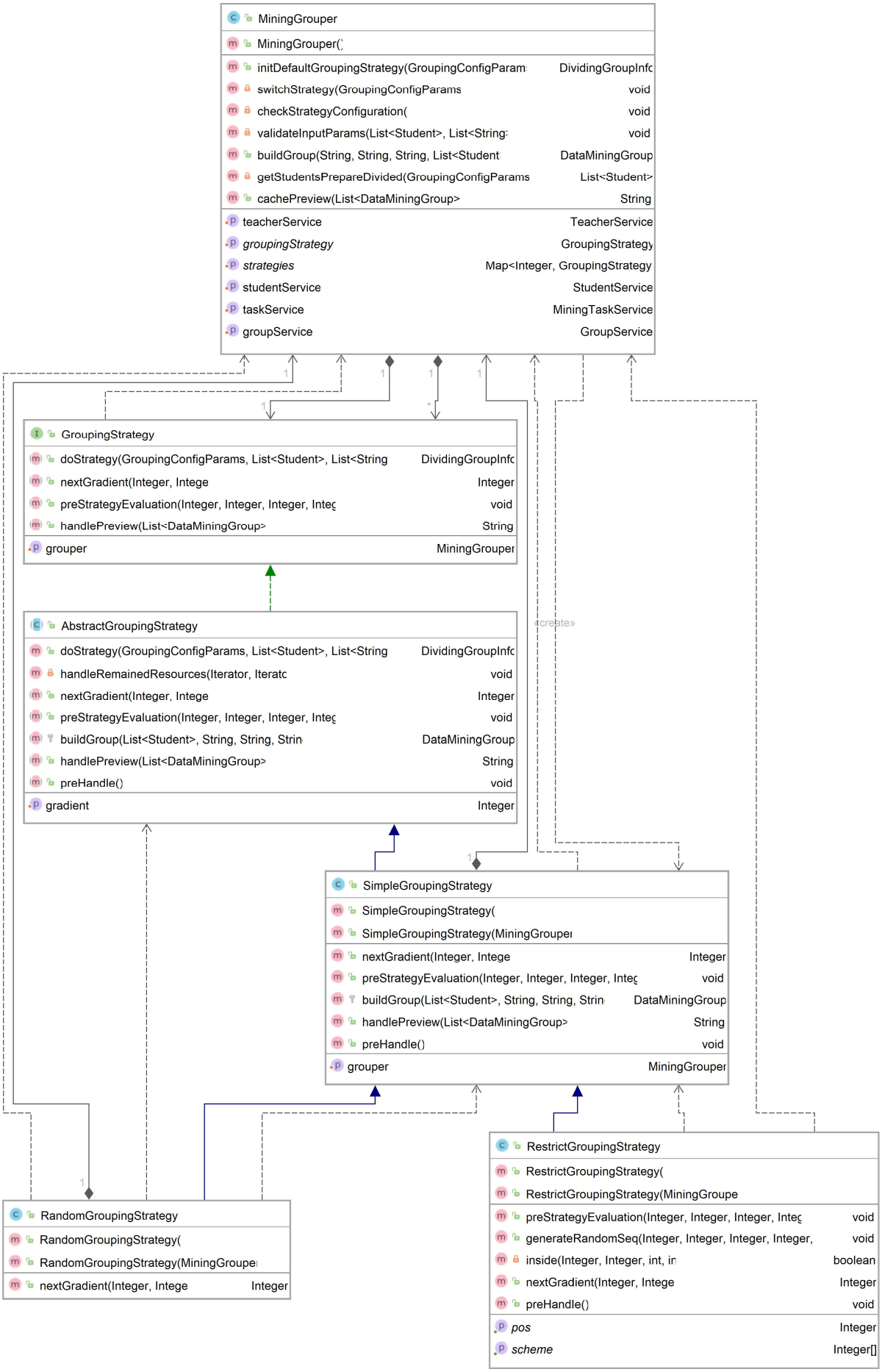
MiningGrouper
MiningGrouper(
initDefaultGroupingStrategy(GroupingConfigParam DividingGroupInfc
switchStrategy(GroupingConfigParams void
checkStrategyConfiguration(void
validateInputParams(List<Student>, List<String void
buildGroup(String, String, String, List<Student DataMiningGroup
getStudentsPrepareDivided(GroupingConfigParams List<Student>
cachePreview(List<DataMiningGroup> String
teacherService TeacherService
groupingStrategy GroupingStrategy
strategies Map<Integer, GroupingStrategy
studentService StudentService
taskService MiningTaskService
groupService GroupService
GroupingStrategy
doStrategy(GroupingConfigParams, List<Student>, List<String DividingGroupInfc
nextGradient(Integer, Intege Integer
preStrategyEvaluation(Integer, Integer, Integer, Intec void
handlePreview(List<DataMiningGroup> String
grouper MiningGrouper
AbstractGroupingStrategy
doStrategy(GroupingConfigParams, List<Student>, List<String DividingGroupInfc
handleRemainedResources(Iterator, Iteratc void
nextGradient(Integer, Intege Integer
preStrategyEvaluation(Integer, Integer, Integer, Intec void
buildGroup(List<Student>, String, String, Strin DataMiningGroup
handlePreview(List<DataMiningGroup> String
preHandle() void
gradient Integer
«create»
SimpleGroupingStrategy
SimpleGroupingStrategy(
SimpleGroupingStrategy(MiningGrouper
nextGradient(Integer, Intege Integer
preStrategyEvaluation(Integer, Integer, Integer, Intec void
buildGroup(List<Student>, String, String, Strin DataMiningGroup
handlePreview(List<DataMiningGroup> String
preHandle() void
grouper MiningGrouper
RestrictGroupingStrategy
RestrictGroupingStrategy(
RestrictGroupingStrategy(MiningGroupe
preStrategyEvaluation(Integer, Integer, Integer, Intec void
generateRandomSeq(Integer, Integer, Integer, Integer, void
inside(Integer, Integer, int, in boolean
nextGradient(Integer, Intege Integer
preHandle() void
pos Integer
scheme Integer[]
RandomGroupingStrategy
RandomGroupingStrategy(
RandomGroupingStrategy(MiningGroupe
nextGradient(Integer, Intege Integer