1. QuizJson название проекта не должно иметь частью технологии. если вместо json будет xml, переименовывать проект?
2. config.properties название переменных должны быть именем существительным и говорить что-то по сути. Например, вместо fileRead - questionsFileName остальные аналогично
3. коментарий в git - add: Completed мало о чем говорит. нужно писать, какое было изменение
4. Quiz.main() не должен содержать никакой логики. просто execute или runTest, etc
5. quizGetQuestions -> getQuizQuestions()
6. List<QuizQuestion> quizQuestionsObjects = null;

...

return quizQuestionsObjects;

избегай потенциального возврата null, потому что потом c NPE намучишься

луче List<QuizQuestion> quizQuestionsObjects = Сollections.emptyList() и NPE не будет

1. BufferedReader нужно закрывать в finally. Cейчас совсем не закрывается - потенциальная утечка памяти

} catch (IOException e) {

e.printStackTrace();

}

лучше, чем ничего, но нужно использовать систему логирования. я на лекциях рассказывал как

1. QuizPassing - название мало о чем говорит, приходиться читать код, чтобы разобраться что делает класс. Please save readers time
2. List quizPassing(List quizGetQuestions) -- List лучше указать какой тип содержит. добавь generics - это поможет найти потенциальные ошибки на этапе компиляции
3. используй логирование вместо System.out.println
4. String filename = "config.properties" - лучше вынести в константу
5. нужно переделать, это процедурное програмирование на обьектно-ориентированном языке

if (request.equals("fileReadLocation")){

return quizProperties.getProperty("pathRead") + quizProperties.getProperty("fileRead");

}

if (request.equals("fileWriteLocation")){

return quizProperties.getProperty("pathWrite") + quizProperties.getProperty("fileWrite");

}

if (request.equals("regexMask")) {

return quizProperties.getProperty("regexMask");

}

else {

return "";

}

1. return ""; - наверное лучше exception выкидывать, ато потом придется долго разбираться почему пустая строка
2. в QuizQuestion все поля public - нарушение инкапсуляции. к тому же, нельзя смешивать представление (System.out) с моделью (QuizQuestion)
3. что если у тебя добавится поле и ты захочешь выводить не на консоль, а в лог файл или в html или во все сразу?
4. в quizVerification() до 7-ми вложеностей!!!!!! максимум 2-3 должно быть. разбей метод на более мелкие методы
5. if (quizUserRating >= quizTotalRating - 20 && quizUserRating >= quizTotalRating - 5 ){ -- 20, 5 вынеси в константы с нужным названием
6. quizQuestions.json поле id тут мне кажется излишним. Поэтому лучше убрать
7. "choice":"A) Копьютер B) Кофеварку C) Холодильник", варианты ответа лучше записавать отдельным масивом, чтобы была возможность их легко смешивать в случайном порядке
8. не понятно зачем answer и key одновремено
9. notifi - notify
10. А где тесты?? (с них нужно начинать)

прогресс есть! поздравляю, молодец!

только нужно еще попрактиковаться с названиями и организацией кода