**Online Quiz Portal Using REST APIs**

**DESCRIPTION:**

To create an online quiz portal with multiple REST APIs where users can browse different quizzes, attempt them, and find their scores and standings.

**DESCRIPTION:**

This project, which aims to create an online quiz application, is based on different REST APIs through which users can log in and take quizzes. After finishing the quiz, they can find their standing and scores based on their accurate responses.

There are two types of users on this quiz portal:

* Admin user
* Participants

**Project Features:**

* Includes a RESTful Web API to perform CRUD operations on Domain objects as per requirement using Spring Boot and MySQL/Oracle database.
* The admin user has a separate API to access the admin portal, which requires authentication with the admin username and password.
* The admin user can update the profile details and change the password after using the access token generated by login.
* The admin user can add questions using the **addQuestions** API.
* The admin user can create a quiz by entering **quizid** and selecting questions with **questionid**.
* The admin user can obtain statistics on total quizzes, questions, and users by using their APIs.
* The admin user can find the users who participated in the quiz along with scores and standings.
* Users can explore various quizzes created by the admin.
* Users can create their profile using new user registration.
* Users can take the quiz and try to answer the questions
* Users can also check if the given answers are correct, as well as their results and positions.

**Tools used:**

* Postman
* Tomcat server
* MySQL

**Admin User Scenario:**

* The admin wants to create an online quiz for the website users.
* The admin creates a set of questions along with their answers.
* Admin opens login API and logs in with the admin username and password.
* Once admin is authenticated, an access token is generated that can be used to add and modify quizzes, questions, and users.
* The admin creates a quiz by providing a name and an id.
* For creating a new quiz, the admin user enters a **quizid** and selects questions from the database using the **questionid**.
* After adding the questions, the admin runs the API and makes the quiz available to its users.
* Once the quiz is released, website users can start taking it.

**Participants Scenario:**

* The user uses the register API to create an account, which provides an access token.
* After registering an account, the user logs in with the access token to take any quiz.
* The user browses various quizzes created by Admin using the **getAllQuizzes** API.
* The user attempts the quiz using **quizid** and gives the possible answers.
* The user views whether the provided answers are right or wrong. The correct answer is highlighted differently so that they are easily identified.
* After completing the quiz, the user checks the scores and compares their standings with other users.

**Pom.xml**

<?xml version="1.0" encoding="UTF-8"?>

<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://maven.apache.org/xsd/maven-4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<parent>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-parent</artifactId>

<version>2.5.2</version>

<relativePath/> <!-- lookup parent from repository -->

</parent>

<groupId>com.example</groupId>

<artifactId>quiz</artifactId>

<version>0.0.1-SNAPSHOT</version>

<name>quiz</name>

<description>Demo project for Spring Boot</description>

<properties>

<java.version>1.7</java.version>

</properties>

<dependencies>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-data-jpa</artifactId>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-web</artifactId>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-devtools</artifactId>

<scope>runtime</scope>

<optional>true</optional>

</dependency>

<dependency>

<groupId>mysql</groupId>

<artifactId>mysql-connector-java</artifactId>

<scope>runtime</scope>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-test</artifactId>

<scope>test</scope>

</dependency>

</dependencies>

<build>

<plugins>

<plugin>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-maven-plugin</artifactId>

</plugin>

</plugins>

</build>

</project>

**QuizApplication.java**

package com.example.quiz;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

import org.springframework.boot.autoconfigure.domain.EntityScan;

import org.springframework.data.jpa.repository.config.EnableJpaRepositories;

@SpringBootApplication

@EntityScan (basePackages = "com.example.quiz.bean")

@EnableJpaRepositories (basePackages = "com.example.quiz.repository")

public class QuizApplication {

public static void main(String[] args) {

SpringApplication.run(QuizApplication.class, args);

}

}

**Login.java**

**package** com.example.quiz.bean;

**import** javax.persistence.Entity;

**import** javax.persistence.Id;

**import** javax.persistence.OneToOne;

**import** javax.persistence.Table;

@Entity

@Table (name = "login")

**public** **class** Login {

@Id

**private** String emailid;

**private** String password;

**private** String typeofuser;

**public** String getEmailid() {

**return** emailid;

}

**public** **void** setEmailid(String emailid) {

**this**.emailid = emailid;

}

**public** String getPassword() {

**return** password;

}

**public** **void** setPassword(String password) {

**this**.password = password;

}

**public** String getTypeofuser() {

**return** typeofuser;

}

**public** **void** setTypeofuser(String typeofuser) {

**this**.typeofuser = typeofuser;

}

@Override

**public** String toString() {

**return** "Login [emailid=" + emailid + ", password=" + password + ", typeofuser=" + typeofuser + "]";

}

**public** Login(String emailid, String password, String typeofuser) {

**super**();

**this**.emailid = emailid;

**this**.password = password;

**this**.typeofuser = typeofuser;

}

**public** Login() {

**super**();

// **TODO** Auto-generated constructor stub

}

}

**Participants.java**

**package** com.example.quiz.bean;

**import** javax.persistence.CascadeType;

**import** javax.persistence.Entity;

**import** javax.persistence.GeneratedValue;

**import** javax.persistence.GenerationType;

**import** javax.persistence.Id;

**import** javax.persistence.JoinColumn;

**import** javax.persistence.OneToOne;

**import** javax.persistence.Table;

@Entity

@Table(name = "participants")

**public** **class** Participants {

@Id

@GeneratedValue (strategy = GenerationType.***IDENTITY***)

**private** **int** pid;

**private** String fname;

**private** String lname;

**private** **int** age;

**private** String emailid;

**private** String password;

**public** String getEmailid() {

**return** emailid;

}

**public** **void** setEmailid(String emailid) {

**this**.emailid = emailid;

}

**public** String getPassword() {

**return** password;

}

**public** **void** setPassword(String password) {

**this**.password = password;

}

**public** **int** getPid() {

**return** pid;

}

**public** Participants(**int** pid, String fname, String lname, **int** age, String emailid, String password) {

**super**();

**this**.pid = pid;

**this**.fname = fname;

**this**.lname = lname;

**this**.age = age;

**this**.emailid = emailid;

**this**.password = password;

}

**public** **void** setPid(**int** pid) {

**this**.pid = pid;

}

**public** String getFname() {

**return** fname;

}

**public** **void** setFname(String fname) {

**this**.fname = fname;

}

**public** String getLname() {

**return** lname;

}

**public** **void** setLname(String lname) {

**this**.lname = lname;

}

**public** **int** getAge() {

**return** age;

}

**public** **void** setAge(**int** age) {

**this**.age = age;

}

@Override

**public** String toString() {

**return** "Participants [pid=" + pid + ", fname=" + fname + ", lname=" + lname + ", age=" + age + ", emailid="

+ emailid + ", password=" + password + "]";

}

**public** Participants() {

**super**();

// **TODO** Auto-generated constructor stub

}

}

**Questions.java**

**package** com.example.quiz.bean;

**import** javax.persistence.Entity;

**import** javax.persistence.GeneratedValue;

**import** javax.persistence.GenerationType;

**import** javax.persistence.Id;

**import** javax.persistence.Table;

@Entity

@Table (name = "questions")

**public** **class** Questions {

@Id

@GeneratedValue (strategy = GenerationType.***AUTO***)

**private** **int** qid;

**private** String question;

**private** String option1;

**private** String option2;

**private** String option3;

**private** String option4;

**private** String correct\_ans;

**public** **int** getQid() {

**return** qid;

}

**public** **void** setQid(**int** qid) {

**this**.qid = qid;

}

**public** String getQuestion() {

**return** question;

}

**public** **void** setQuestion(String question) {

**this**.question = question;

}

**public** String getOption1() {

**return** option1;

}

**public** **void** setOption1(String option1) {

**this**.option1 = option1;

}

**public** String getOption2() {

**return** option2;

}

**public** **void** setOption2(String option2) {

**this**.option2 = option2;

}

**public** String getOption3() {

**return** option3;

}

**public** **void** setOption3(String option3) {

**this**.option3 = option3;

}

**public** String getOption4() {

**return** option4;

}

**public** **void** setOption4(String option4) {

**this**.option4 = option4;

}

**public** String getCorrect\_ans() {

**return** correct\_ans;

}

**public** **void** setCorrect\_ans(String correct\_ans) {

**this**.correct\_ans = correct\_ans;

}

@Override

**public** String toString() {

**return** "Questions [qid=" + qid + ", question=" + question + ", option1=" + option1 + ", option2=" + option2

+ ", option3=" + option3 + ", option4=" + option4 + ", correct\_ans=" + correct\_ans + "]";

}

**public** Questions(**int** qid, String question, String option1, String option2, String option3, String option4,

String correct\_ans) {

**super**();

**this**.qid = qid;

**this**.question = question;

**this**.option1 = option1;

**this**.option2 = option2;

**this**.option3 = option3;

**this**.option4 = option4;

**this**.correct\_ans = correct\_ans;

}

**public** Questions() {

**super**();

// **TODO** Auto-generated constructor stub

}

}

**Quizes.java**

**package** com.example.quiz.bean;

**import** javax.persistence.Entity;

**import** javax.persistence.GeneratedValue;

**import** javax.persistence.GenerationType;

**import** javax.persistence.Id;

@Entity

**public** **class** Quizes {

@Id

@GeneratedValue (strategy = GenerationType.***IDENTITY***)

**private** **int** qzid;

**private** String title;

**private** **int** qid;

**public** **int** getQzid() {

**return** qzid;

}

**public** **void** setQzid(**int** qzid) {

**this**.qzid = qzid;

}

**public** String getTitle() {

**return** title;

}

**public** **void** setTitle(String title) {

**this**.title = title;

}

**public** **int** getQid() {

**return** qid;

}

**public** **void** setQid(**int** qid) {

**this**.qid = qid;

}

@Override

**public** String toString() {

**return** "Quizes [qzid=" + qzid + ", title=" + title + ", qid=" + qid + "]";

}

**public** Quizes(**int** qzid, String title, **int** qid) {

**super**();

**this**.qzid = qzid;

**this**.title = title;

**this**.qid = qid;

}

**public** Quizes() {

**super**();

// **TODO** Auto-generated constructor stub

}

}

**Result.java**

**package** com.example.quiz.bean;

**import** javax.persistence.Entity;

**import** javax.persistence.Id;

**import** javax.persistence.Table;

@Entity

@Table(name = "result")

**public** **class** Result {

@Id

**private** String emailid;

**private** **int** score;

**private** String title;

**private** **int** standing\_position;

**public** String getEmailid() {

**return** emailid;

}

**public** **void** setEmailid(String emailid) {

**this**.emailid = emailid;

}

**public** **int** getScore() {

**return** score;

}

**public** **void** setScore(**int** score) {

**this**.score = score;

}

**public** String getTitle() {

**return** title;

}

**public** **void** setTitle(String title) {

**this**.title = title;

}

**public** **int** getStanding\_position() {

**return** standing\_position;

}

**public** **void** setStanding\_position(**int** standing\_position) {

**this**.standing\_position = standing\_position;

}

@Override

**public** String toString() {

**return** "Result of the participant " + emailid + " of subject "+title+" : score = " + score + ", rank = " + standing\_position ;

}

**public** Result(String emailid, **int** score, String title, **int** standing\_position) {

**super**();

**this**.emailid = emailid;

**this**.score = score;

**this**.title = title;

**this**.standing\_position = standing\_position;

}

**public** Result() {

**super**();

// **TODO** Auto-generated constructor stub

}

}

**AdminController.java**

**package** com.example.quiz.controller;

**import** java.util.List;

**import** org.springframework.beans.factory.annotation.Autowired;

**import** org.springframework.http.MediaType;

**import** org.springframework.web.bind.annotation.GetMapping;

**import** org.springframework.web.bind.annotation.PathVariable;

**import** org.springframework.web.bind.annotation.PostMapping;

**import** org.springframework.web.bind.annotation.RequestBody;

**import** org.springframework.web.bind.annotation.RequestMapping;

**import** org.springframework.web.bind.annotation.RestController;

**import** com.example.quiz.bean.Login;

**import** com.example.quiz.bean.Questions;

**import** com.example.quiz.bean.Quizes;

**import** com.example.quiz.bean.Result;

**import** com.example.quiz.service.LoginService;

**import** com.example.quiz.service.QuestionsService;

**import** com.example.quiz.service.QuizesService;

**import** com.example.quiz.service.ResultService;

@RestController

@RequestMapping("admin")

**public** **class** AdminController {

@Autowired

LoginService loginService;

@Autowired

QuestionsService questionsService;

@Autowired

QuizesService quizesService;

@Autowired

ResultService resultService;

//

@GetMapping (value = "print")

**public** String print() {

**return** "Welcome to the Quiz rest API --> ADMIN Controller";

}

//

@PostMapping (value = "signIn", consumes = MediaType.***APPLICATION\_JSON\_VALUE***)

**public** String signIn(@RequestBody Login login) {

**return** loginService.signIn(login);

}

//

@PostMapping (value = "changePassword", consumes = MediaType.***APPLICATION\_JSON\_VALUE***)

**public** String changePassword(@RequestBody Login login) {

**return** loginService.updatePassword(login);

}

//

@PostMapping (value = "addQuestion", consumes = MediaType.***APPLICATION\_JSON\_VALUE***)

**public** String addQuestion(@RequestBody Questions ques) {

**return** questionsService.storeQuestion(ques);

}

//

@GetMapping (value = "findQuestionById/{qid}", produces = MediaType.***APPLICATION\_JSON\_VALUE***)

**public** Questions findQuestionById(@PathVariable("qid") **int** qid) {

**return** questionsService.findQuestion(qid);

}

//

@PostMapping(value = "createQuiz", consumes = MediaType.***APPLICATION\_JSON\_VALUE***)

**public** String createQuiz(@RequestBody Quizes quiz) {

**return** quizesService.addQuiz(quiz);

}

//

@GetMapping(value = "checkUserResult", produces = MediaType.***APPLICATION\_JSON\_VALUE***)

**public** List<Result> checkResult(){

**return** resultService.getAllResult();

}

}

**ParticipantsController.java**

**package** com.example.quiz.controller;

**import** java.util.List;

**import** org.springframework.beans.factory.annotation.Autowired;

**import** org.springframework.http.MediaType;

**import** org.springframework.web.bind.annotation.GetMapping;

**import** org.springframework.web.bind.annotation.PathVariable;

**import** org.springframework.web.bind.annotation.PostMapping;

**import** org.springframework.web.bind.annotation.RequestBody;

**import** org.springframework.web.bind.annotation.RequestMapping;

**import** org.springframework.web.bind.annotation.RestController;

**import** com.example.quiz.bean.Login;

**import** com.example.quiz.bean.Participants;

**import** com.example.quiz.bean.Quizes;

**import** com.example.quiz.bean.Result;

**import** com.example.quiz.service.LoginService;

**import** com.example.quiz.service.ParticipantsService;

**import** com.example.quiz.service.QuizesService;

**import** com.example.quiz.service.ResultService;

// http://localhost:9191/participants

@RestController

@RequestMapping("participants")

**public** **class** ParticipantsController {

@Autowired

LoginService loginService;

@Autowired

ParticipantsService participantsService;

@Autowired

QuizesService quizesService;

@Autowired

ResultService resultService;

//

@PostMapping(value = "signup", consumes = MediaType.***APPLICATION\_JSON\_VALUE***)

**public** String signup(@RequestBody Participants participants) {

**return** participantsService.addParticipant(participants);

}

@GetMapping(value = "takeQuiz/{id}")

**public** String takeQuiz(@PathVariable("id") Quizes quiz) {

**return** quizesService.findQuiz(quiz);

}

//

@PostMapping (value = "signIn", consumes = MediaType.***APPLICATION\_JSON\_VALUE***)

**public** String signIn(@RequestBody Login login) {

**return** loginService.signInUser(login);

}

//

@GetMapping(value = "checkAllQuizDetails", produces = MediaType.***APPLICATION\_JSON\_VALUE***)

**public** List<Quizes> checkAllQuizDetails(){

**return** quizesService.getAllQuizes();

}

//

@GetMapping(value = "checkResult", produces = MediaType.***APPLICATION\_JSON\_VALUE***)

**public** List<Result> checkResult(){

**return** resultService.getAllResult();

}

//

@PostMapping(value = "checkResult1", consumes = MediaType.***APPLICATION\_JSON\_VALUE***)

**public** String checkResult1(@RequestBody Result result) {

**return** resultService.findByIdUser(result);

}

}

**LoginRepository.java**

**package** com.example.quiz.repository;

**import** org.springframework.data.jpa.repository.JpaRepository;

**import** org.springframework.stereotype.Repository;

**import** com.example.quiz.bean.Login;;

@Repository

**public** **interface** LoginRepository **extends** JpaRepository<Login, String>{

}

**ParticipantsRepository.java**

**package** com.example.quiz.repository;

**import** org.springframework.data.jpa.repository.JpaRepository;

**import** org.springframework.stereotype.Repository;

//import com.example.quiz.bean;

**import** com.example.quiz.bean.Participants;

@Repository

**public** **interface** ParticipantsRepository **extends** JpaRepository<Participants, Integer> {

}

**QuestionsRepository.java**

**package** com.example.quiz.repository;

**import** org.springframework.data.jpa.repository.JpaRepository;

**import** org.springframework.stereotype.Repository;

**import** com.example.quiz.bean.Questions;

@Repository

**public** **interface** QuestionsRepository **extends** JpaRepository<Questions, Integer> {

}

**QuizesRepository.java**

**package** com.example.quiz.repository;

**import** java.util.Optional;

**import** org.springframework.data.jpa.repository.JpaRepository;

**import** org.springframework.stereotype.Repository;

**import** com.example.quiz.bean.Quizes;

@Repository

**public** **interface** QuizesRepository **extends** JpaRepository<Quizes, Integer>{

}

**ResultRepository.java**

**package** com.example.quiz.repository;

**import** org.springframework.data.jpa.repository.JpaRepository;

**import** org.springframework.stereotype.Repository;

**import** com.example.quiz.bean.Result;

@Repository

**public** **interface** ResultRepository **extends** JpaRepository<Result, String> {

}

**LoginService.java**

**package** com.example.quiz.service;

**import** java.util.Optional;

**import** org.springframework.beans.factory.annotation.Autowired;

**import** org.springframework.stereotype.Service;

**import** com.example.quiz.bean.Login;

**import** com.example.quiz.repository.LoginRepository;

@Service

**public** **class** LoginService {

@Autowired

LoginRepository loginRepository;

**public** String signIn(Login login) {

Optional<Login> op = loginRepository.findById(login.getEmailid());

**if**(op.isPresent() && login.getPassword().equals("uma@123")) {

Login user = op.get();

**return** "Admin logged in successfully";

}**else** {

**return** "Wrong credentials, Try again";

}

}

**public** String updatePassword(Login login) {

Optional<Login> op = loginRepository.findById(login.getEmailid());

**if**(op.isPresent()) {

Login user = op.get();

user.setPassword(login.getPassword());

loginRepository.saveAndFlush(user);

**return** "Password updated successfully";

}**else** {

**return** "Password didn't update";

}

}

**public** String signInUser(Login login) {

Optional<Login> op = loginRepository.findById(login.getEmailid());

**if**(op.isPresent() && ((login.getEmailid().equals("uma@gmail.com") && login.getPassword().equals("uma@143")) || (login.getEmailid().equals("vyshu@gmail.com") && login.getPassword().equals("vyshu@123")))) {

Login user = op.get();

**return** "Participant logged in successfully";

}**else** {

**return** "Wrong credentials, Try again";

}

}

}

**ParticipantsService.java**

**package** com.example.quiz.service;

**import** java.util.Optional;

**import** org.springframework.beans.factory.annotation.Autowired;

**import** org.springframework.stereotype.Service;

**import** com.example.quiz.bean.Participants;

**import** com.example.quiz.repository.ParticipantsRepository;

@Service

**public** **class** ParticipantsService {

@Autowired

ParticipantsRepository participantsRepository;

**public** String addParticipant(Participants participants) {

Optional<Participants> op = participantsRepository.findById(participants.getPid());

**if**(op.isPresent()) {

**return** "Participant id must be unique";

}**else** {

participantsRepository.save(participants);

**return** "Participant details added successfully";

}

}

}

**QuestionsService.java**

**package** com.example.quiz.service;

**import** java.util.Optional;

**import** org.springframework.beans.factory.annotation.Autowired;

**import** org.springframework.stereotype.Service;

**import** com.example.quiz.bean.Questions;

**import** com.example.quiz.repository.QuestionsRepository;

@Service

**public** **class** QuestionsService {

@Autowired

QuestionsRepository questionsRepository;

**public** String storeQuestion(Questions questions) {

Optional<Questions> op = questionsRepository.findById(questions.getQid());

**if**(op.isPresent()) {

**return** "Question id must be unique";

}**else** {

questionsRepository.save(questions);

**return** "Question added successfully";

}

}

**public** Questions findQuestion(**int** qid) {

Optional<Questions> op = questionsRepository.findById(qid);

**if**(op.isPresent()) {

Questions question = op.get();

**return** question;

}**else** {

**return** **null**;

}

}

}

**QuizesService.java**

**package** com.example.quiz.service;

**import** java.util.List;

**import** java.util.Optional;

**import** org.springframework.beans.factory.annotation.Autowired;

**import** org.springframework.stereotype.Service;

**import** com.example.quiz.repository.QuizesRepository;

**import** com.example.quiz.bean.Quizes;

@Service

**public** **class** QuizesService {

@Autowired

QuizesRepository quizesRepository;

**public** String addQuiz(Quizes quiz) {

Optional<Quizes> op = quizesRepository.findById(quiz.getQzid());

**if**(op.isPresent()) {

**return** "Quiz id must be unique";

}**else** {

quizesRepository.save(quiz);

**return** "Quiz added succesfully";

}

}

**public** String findQuiz(Quizes quiz) {

Optional<Quizes> op = quizesRepository.findById(quiz.getQzid());

**if**(op.isPresent()) {

Quizes q = op.get();

**return** q.toString();

}**else** {

**return** "There is no quiz for this quiz Id";

}

}

**public** List<Quizes> getAllQuizes(){

**return** quizesRepository.findAll();

}

}

**ResultService.java**

**package** com.example.quiz.service;

**import** java.util.List;

**import** java.util.Optional;

**import** org.springframework.beans.factory.annotation.Autowired;

**import** org.springframework.stereotype.Service;

**import** com.example.quiz.bean.Result;

**import** com.example.quiz.repository.ResultRepository;

@Service

**public** **class** ResultService {

@Autowired

ResultRepository resultRepository;

**public** String findByIdUser(Result result) {

Optional<Result> op = resultRepository.findById(result.getEmailid());

**if**(op.isEmpty()) {

**return** "You have not attempted any test";

}**else** {

Result r = op.get();

**return** r.toString();

}

}

**public** List<Result> getAllResult(){

**return** resultRepository.findAll();

}

}

**application.properties**

spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver

spring.datasource.url=jdbc:mysql://localhost:3306/quizdb

spring.datasource.username=root

spring.datasource.password=Uma@1472#la

server.port = 8080

spring.jpa.hibernate.ddl-auto=create

**QuizApplicationTests.java**

package com.example.quiz;

import org.junit.jupiter.api.Test;

import org.springframework.boot.test.context.SpringBootTest;

@SpringBootTest

class QuizApplicationTests {

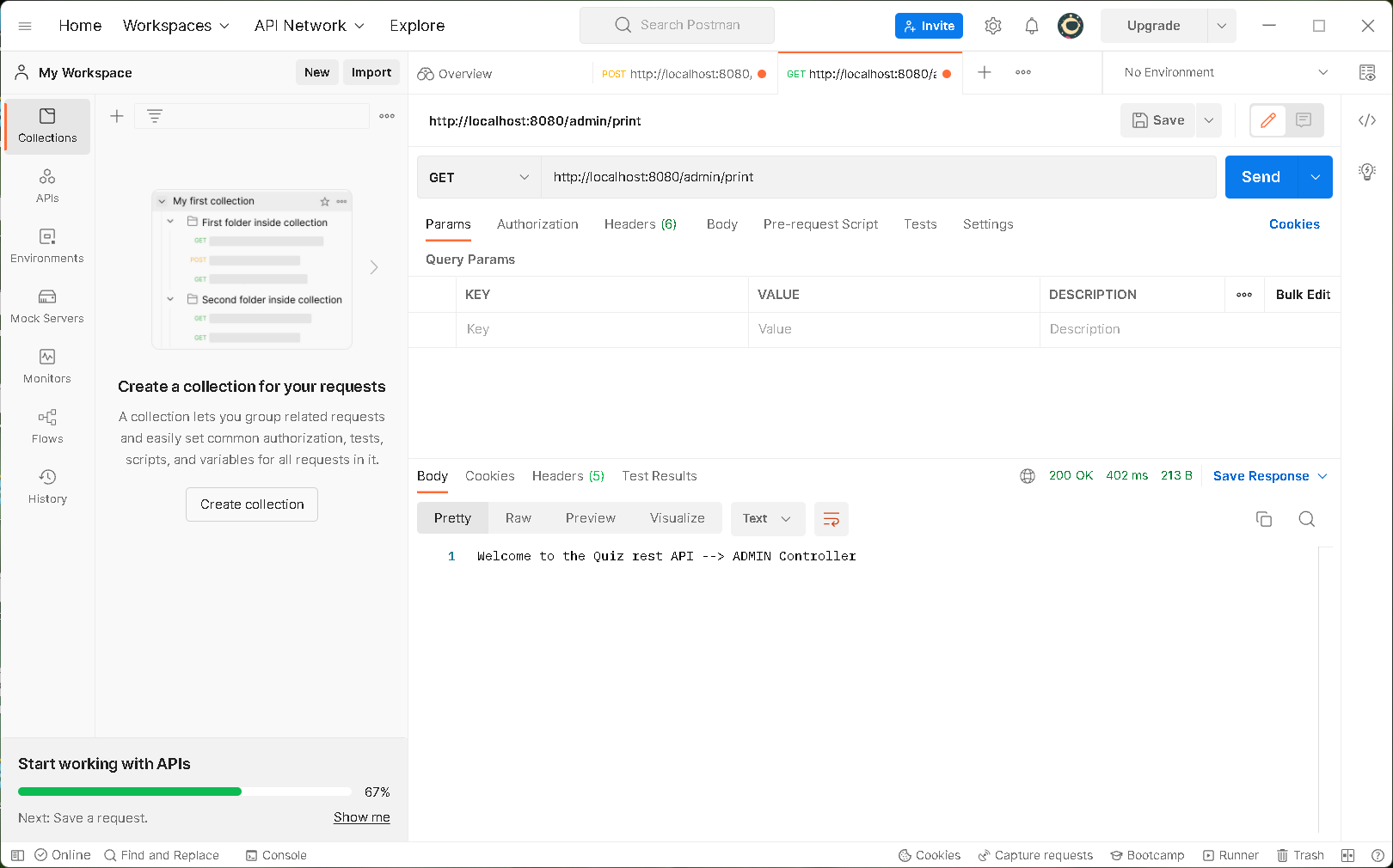
@Test

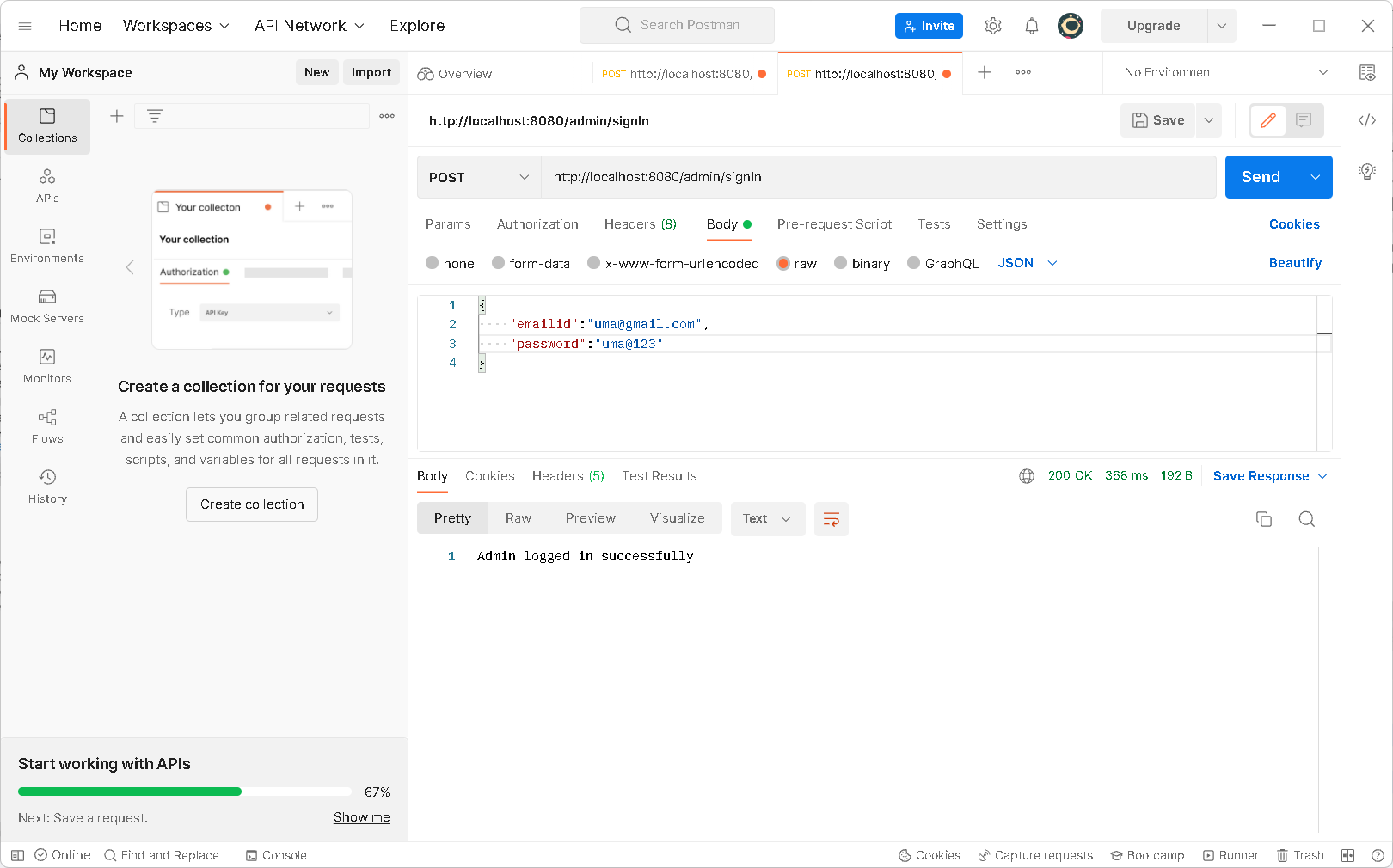
void contextLoads() {

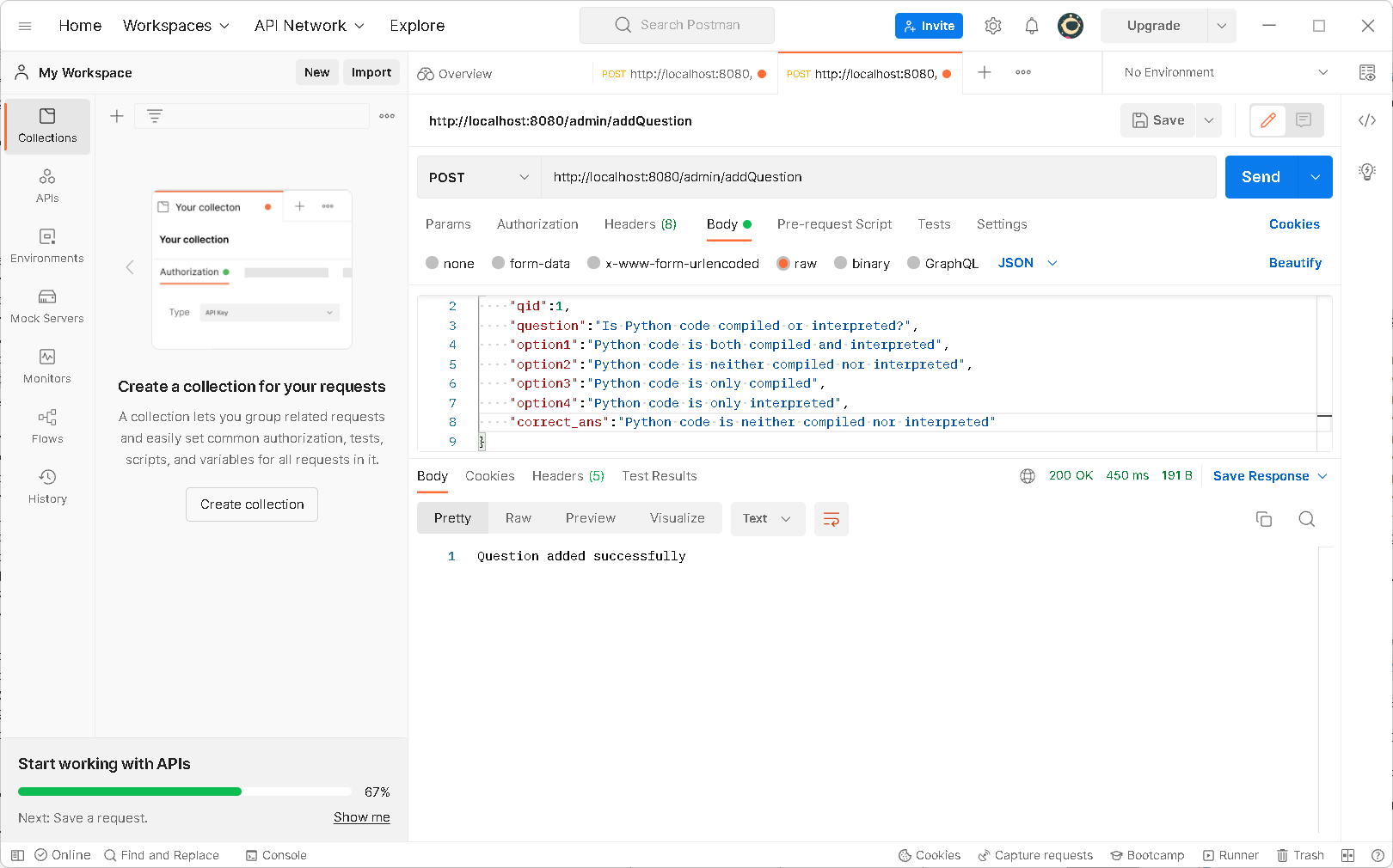
}

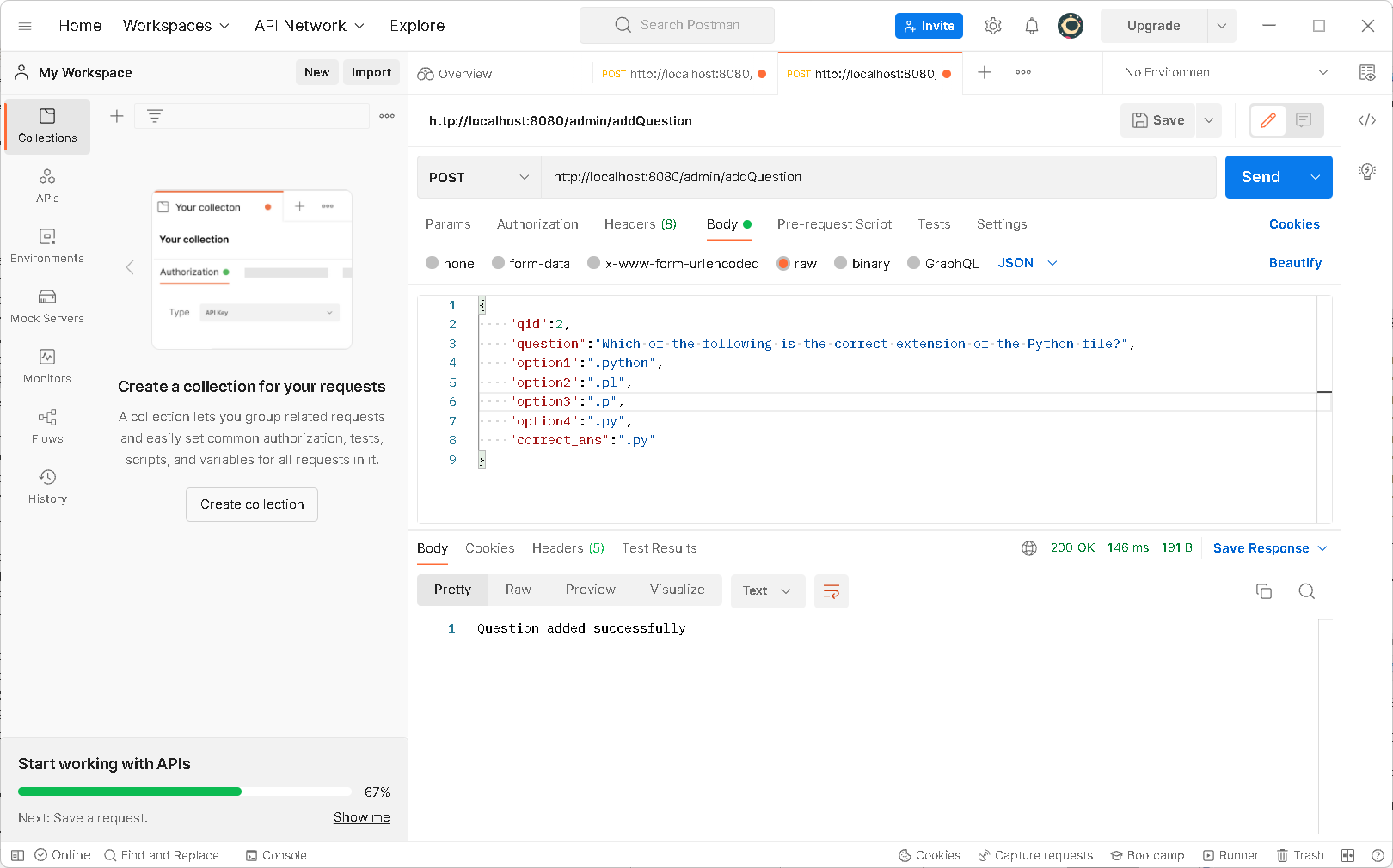
}

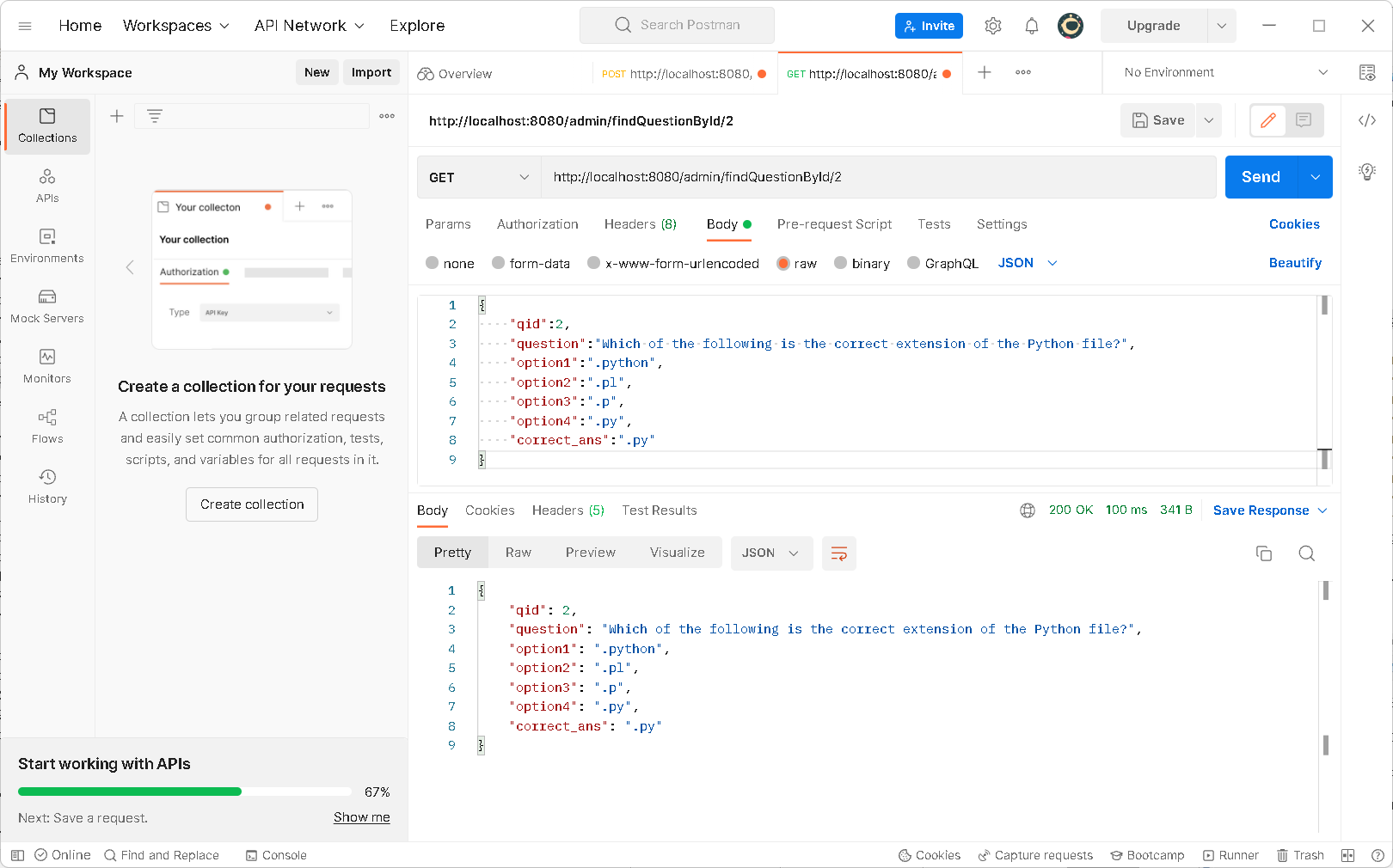
**Output:**

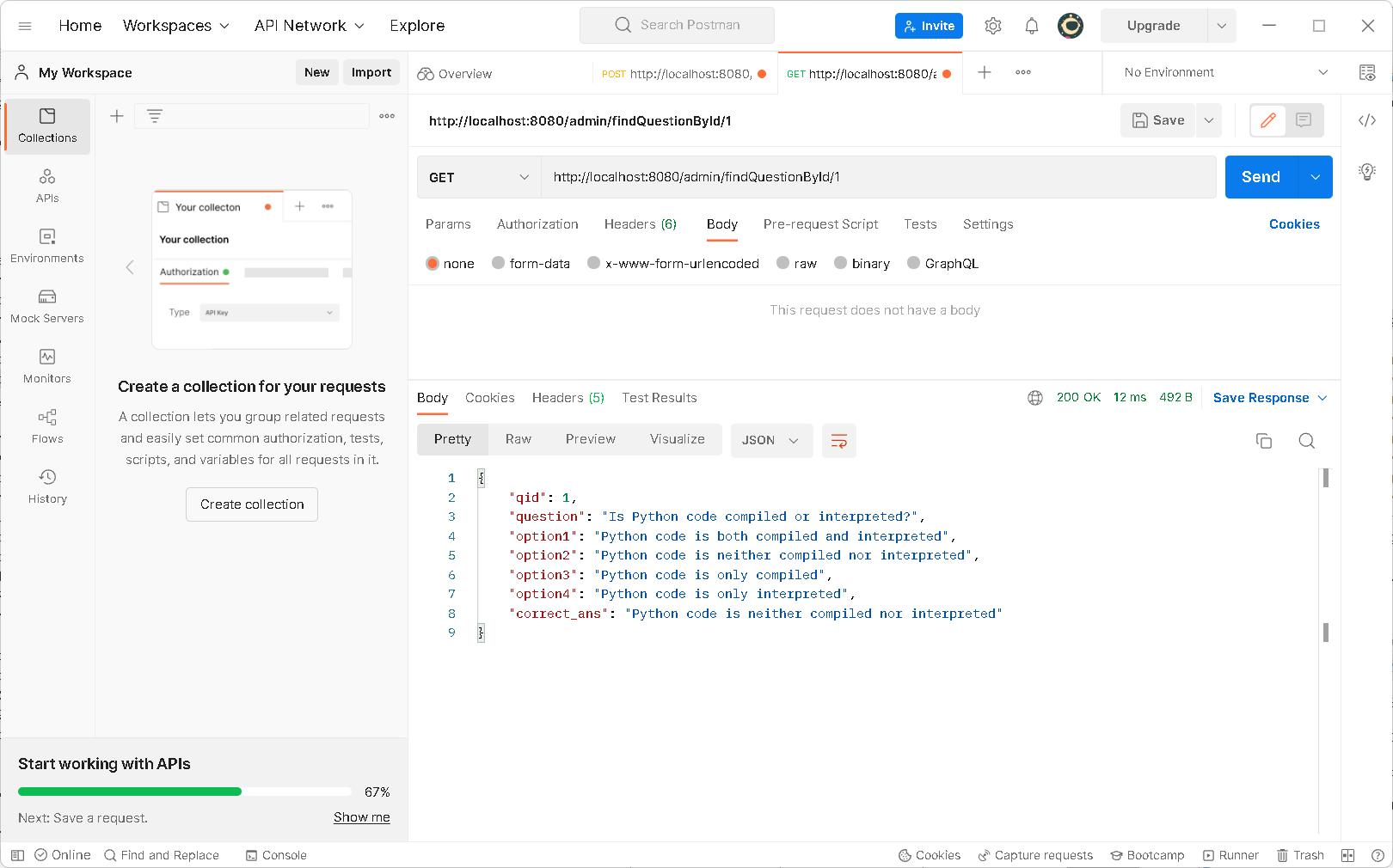
****

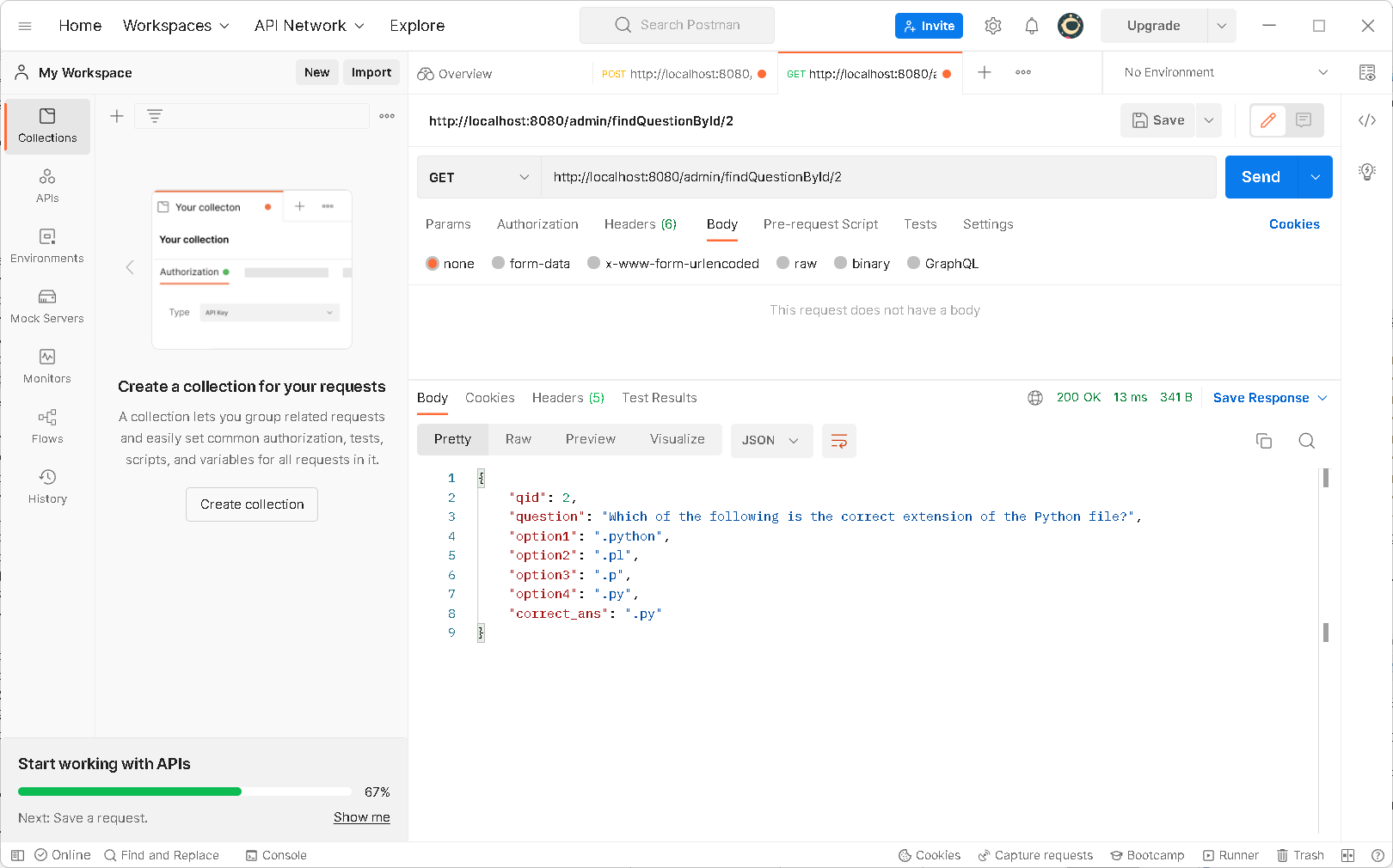
****

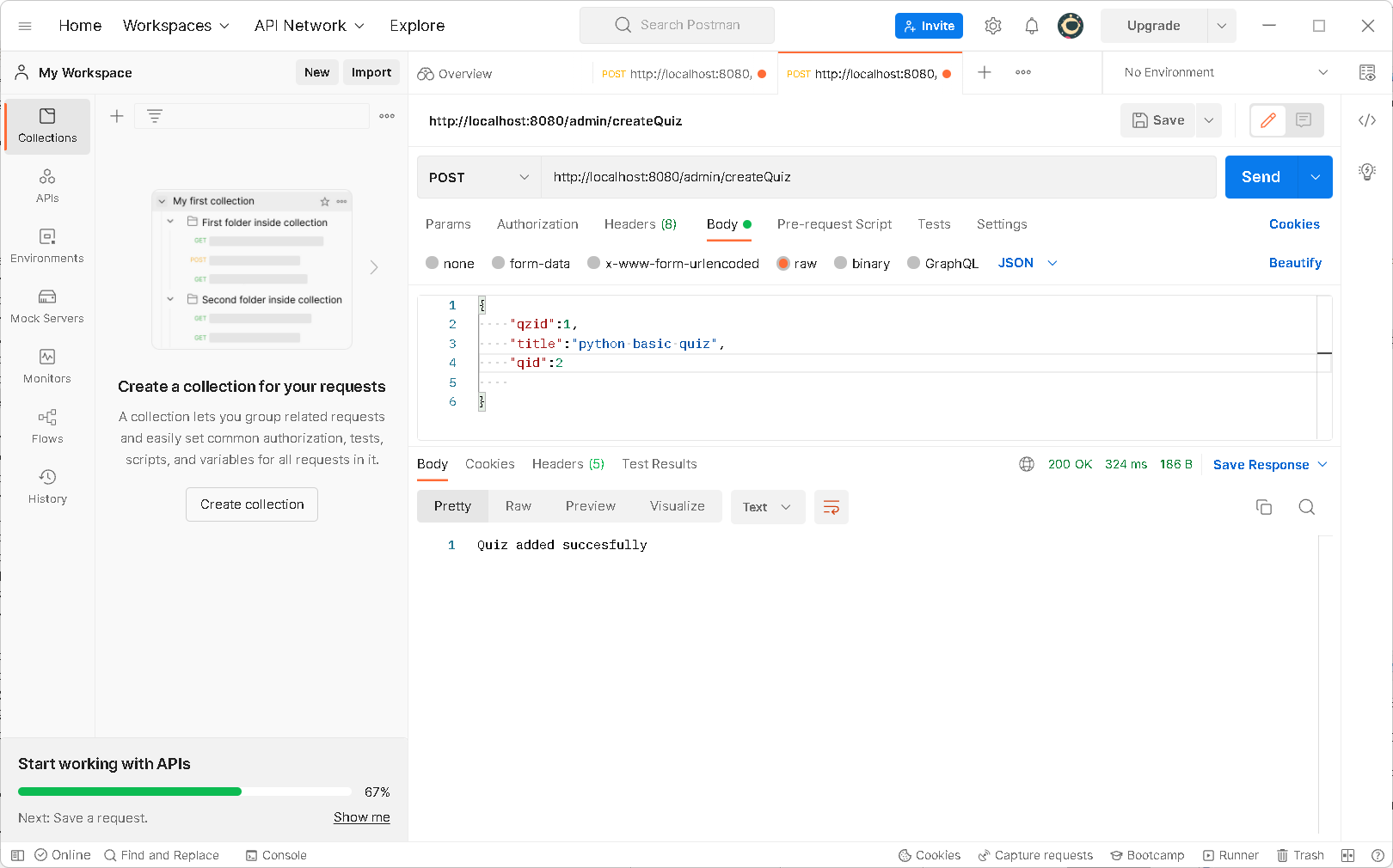
****

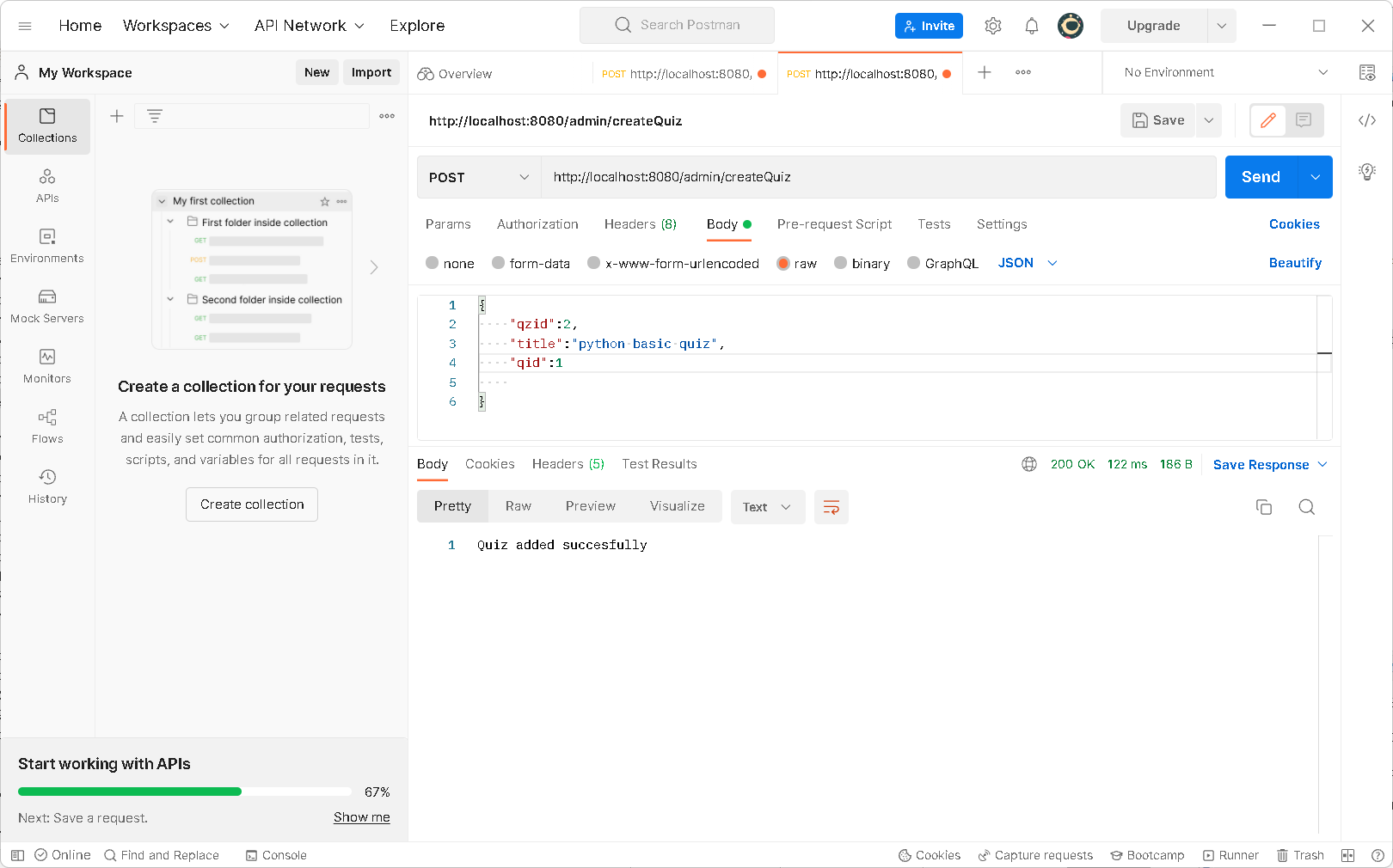
****

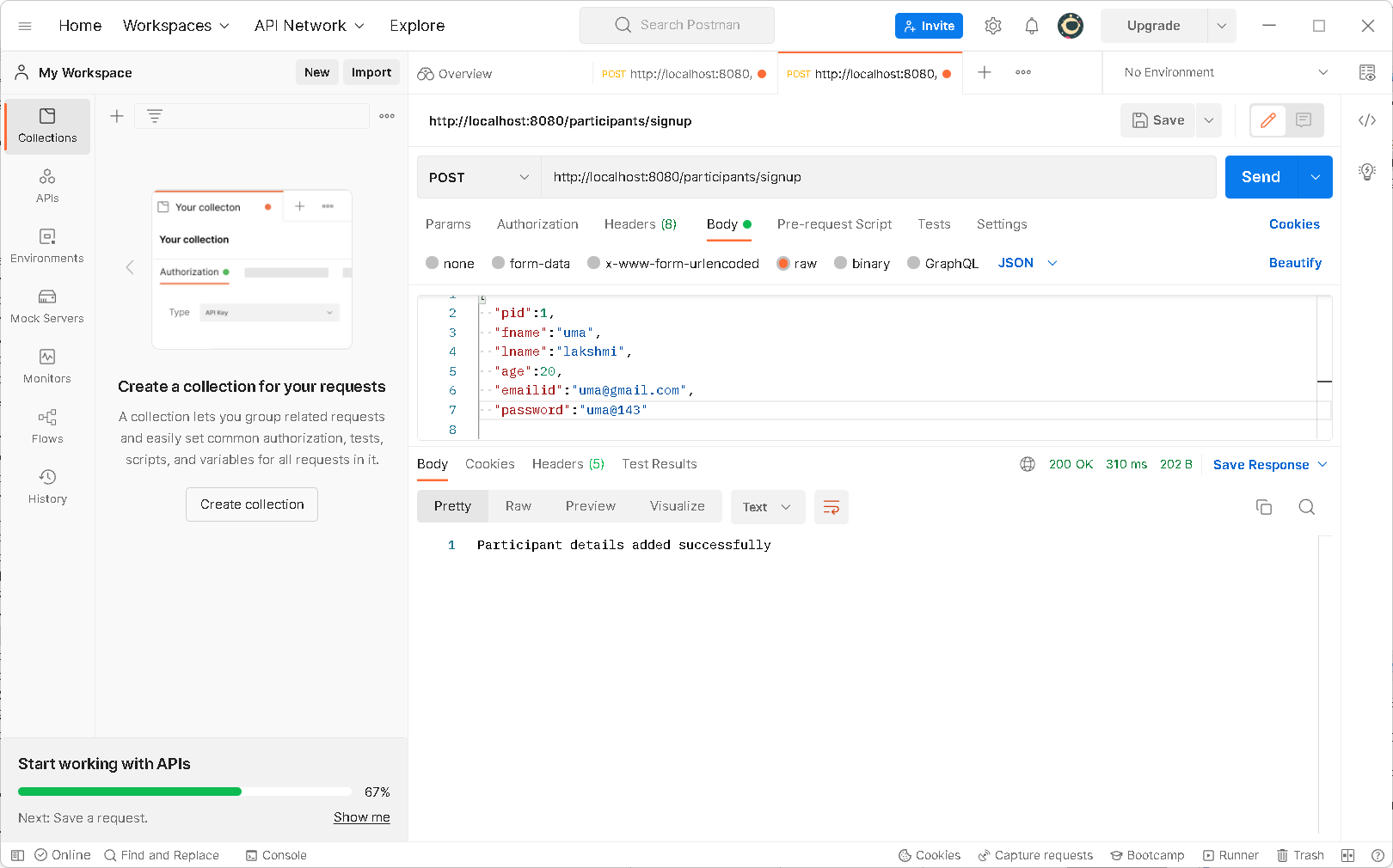
****

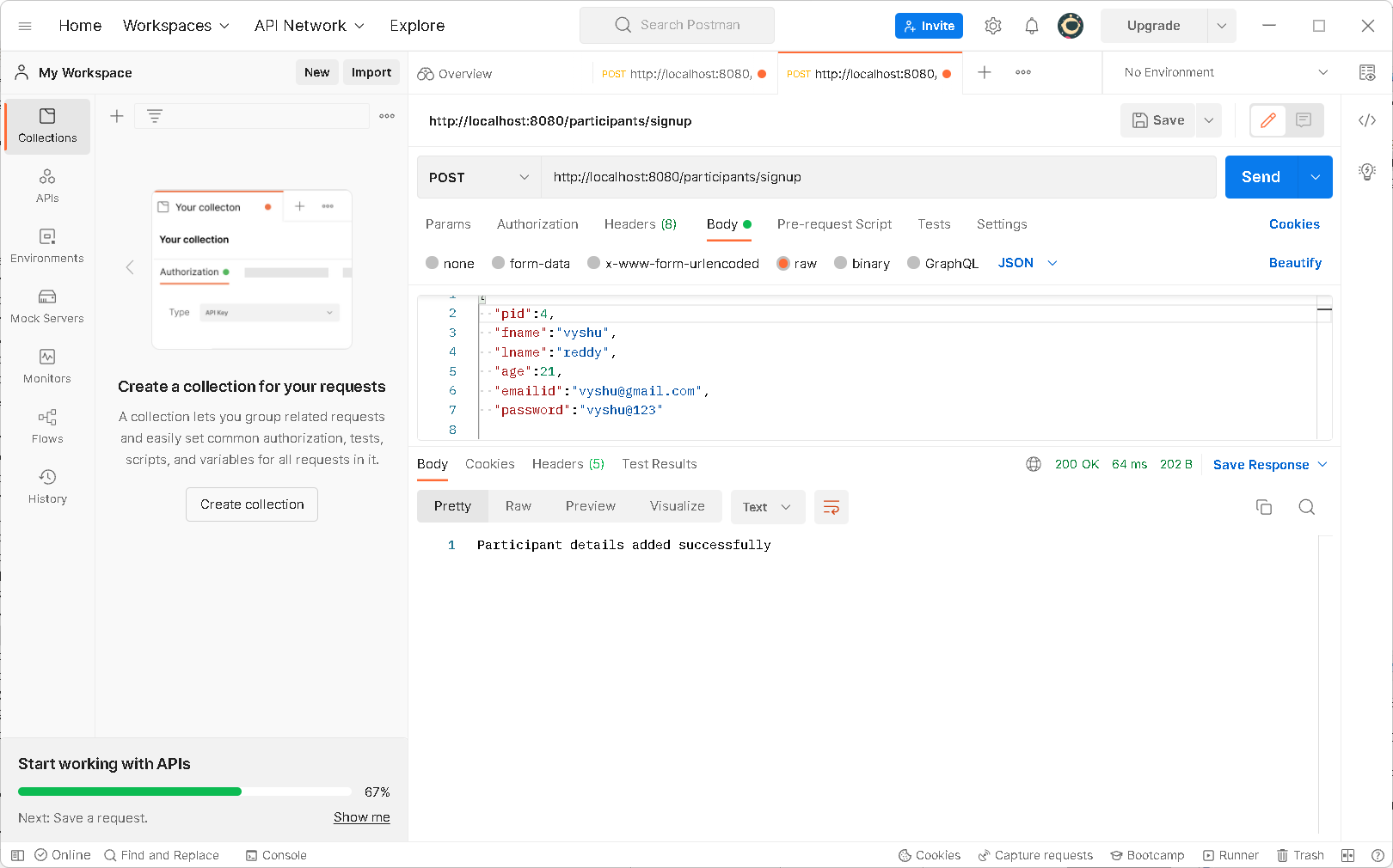
****

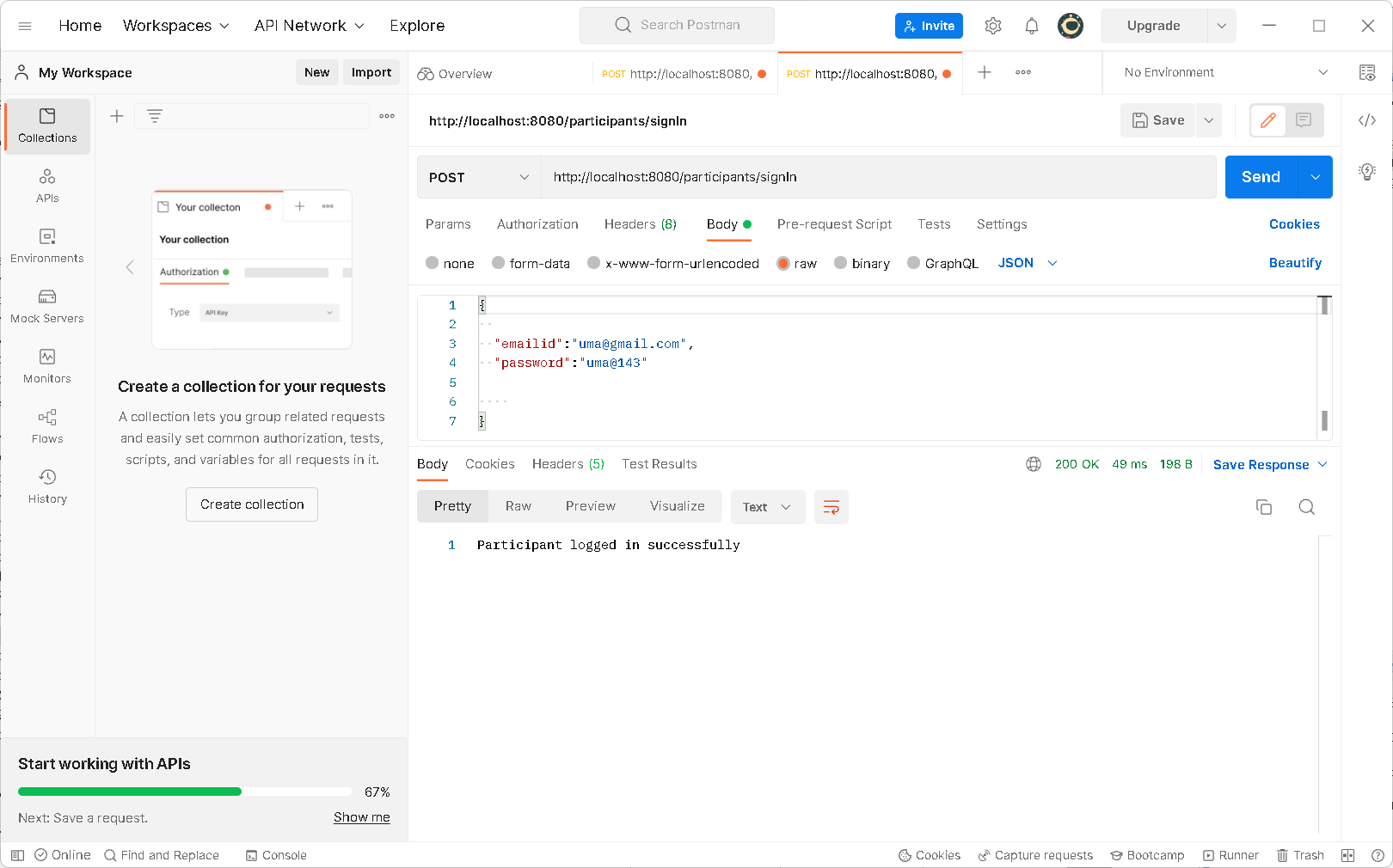
****

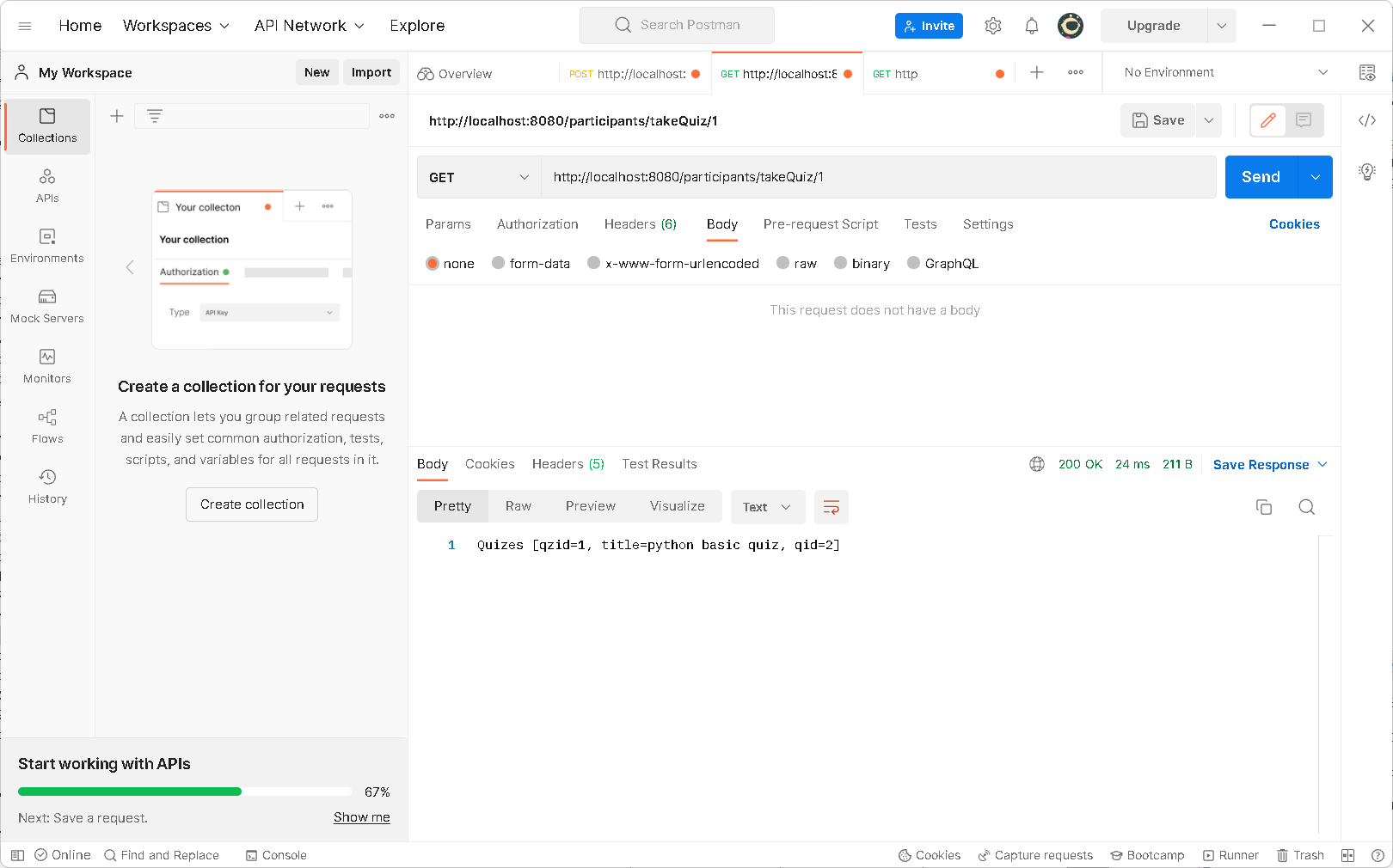
****

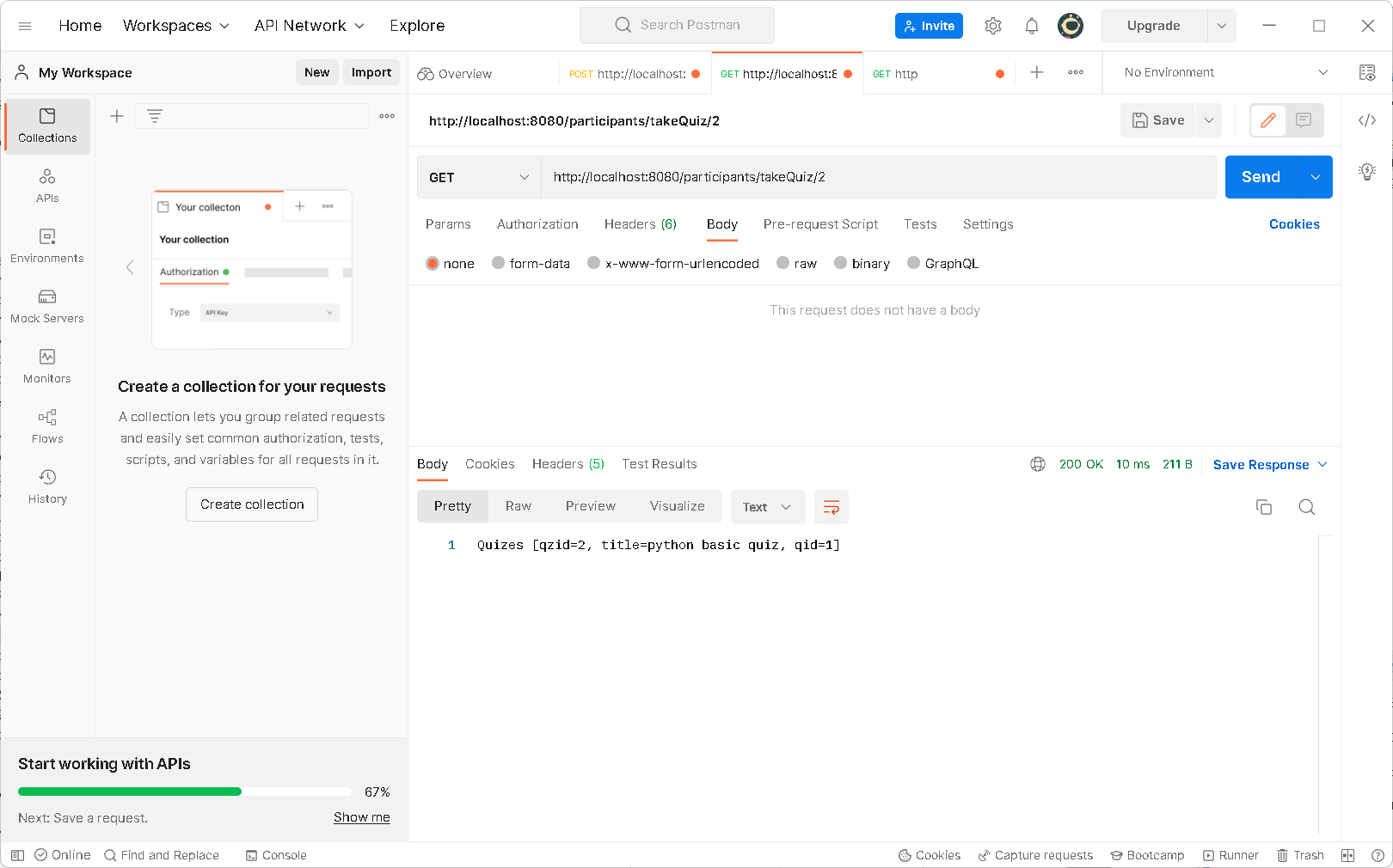
****

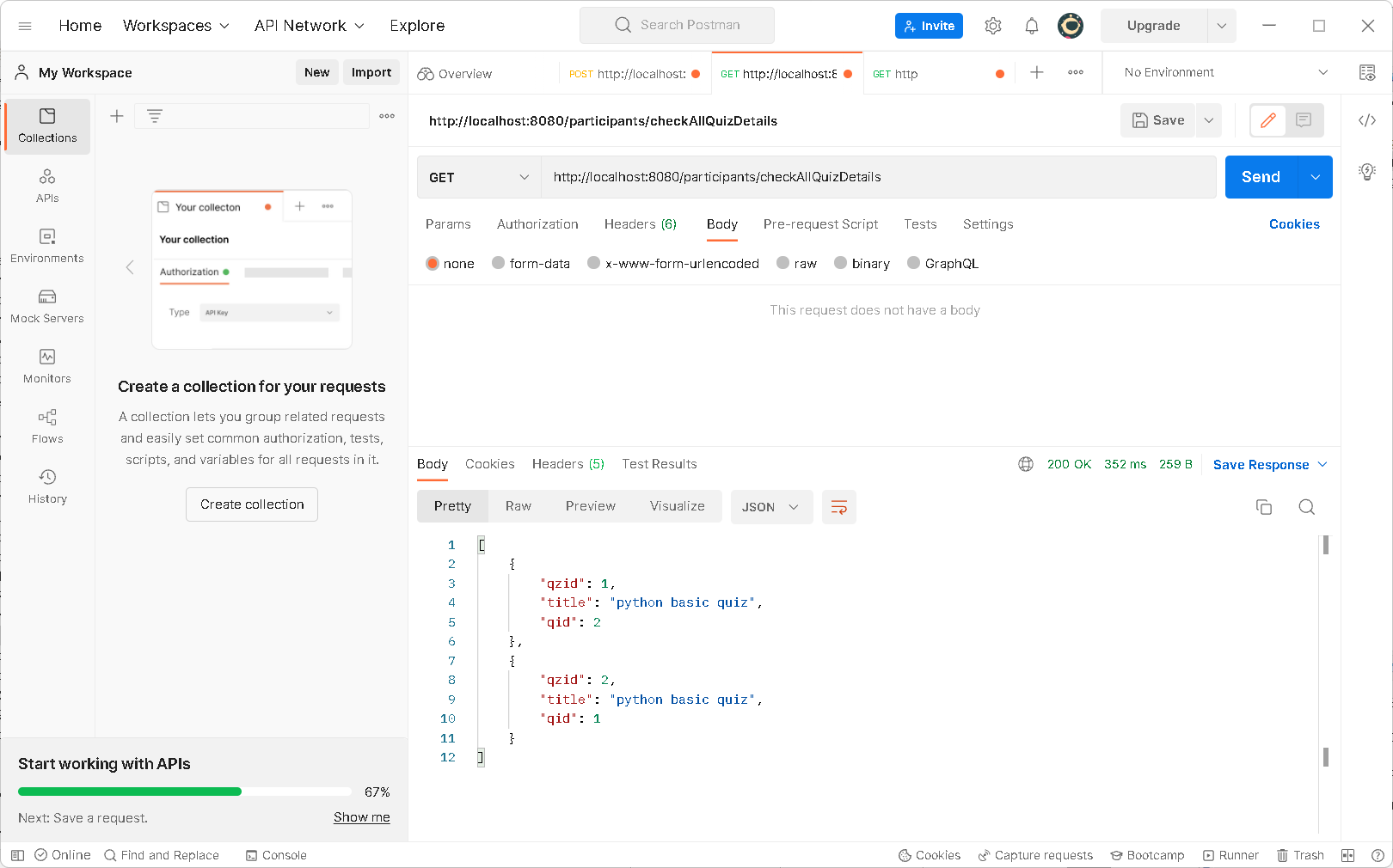
****

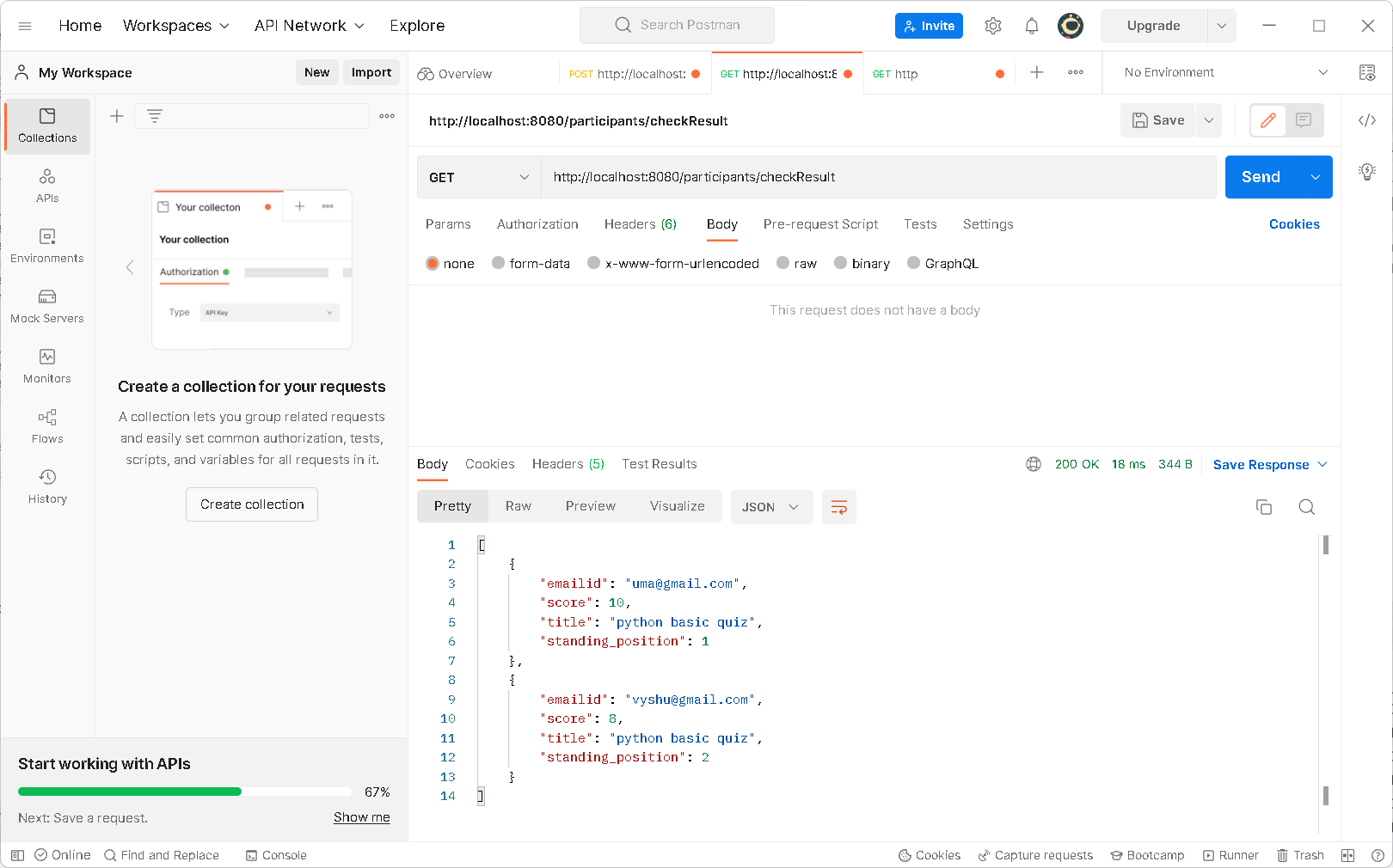
****

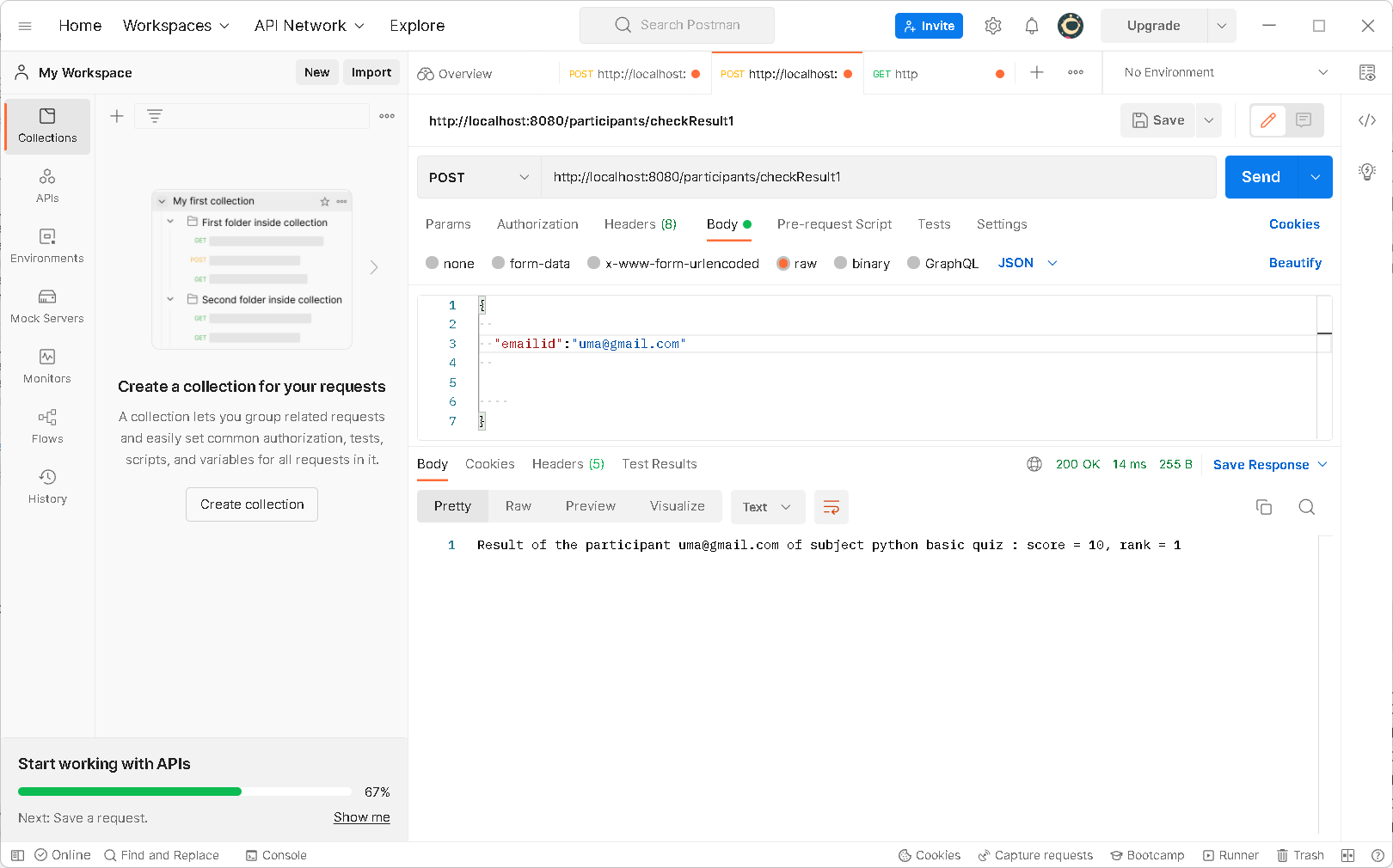
****

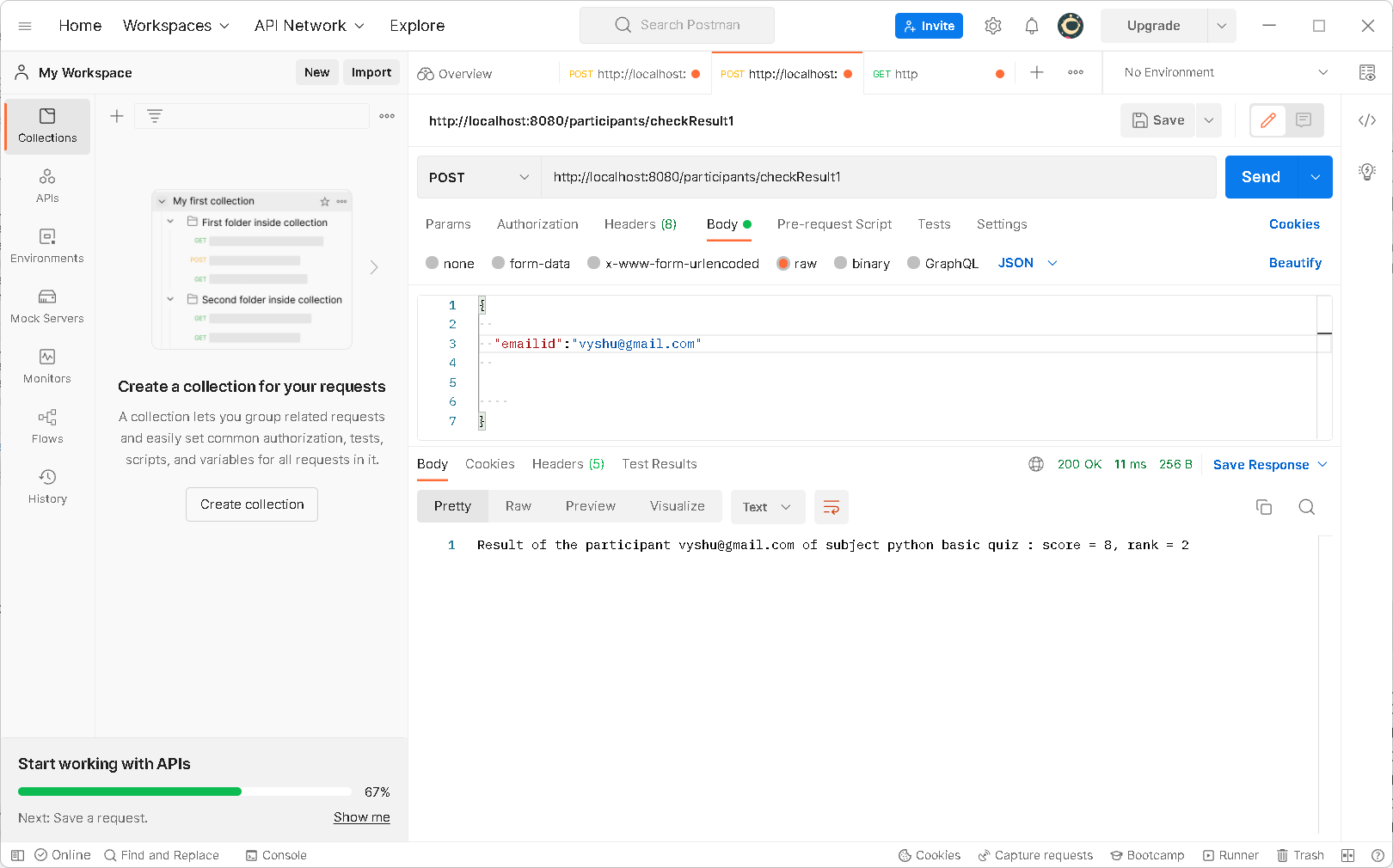
****

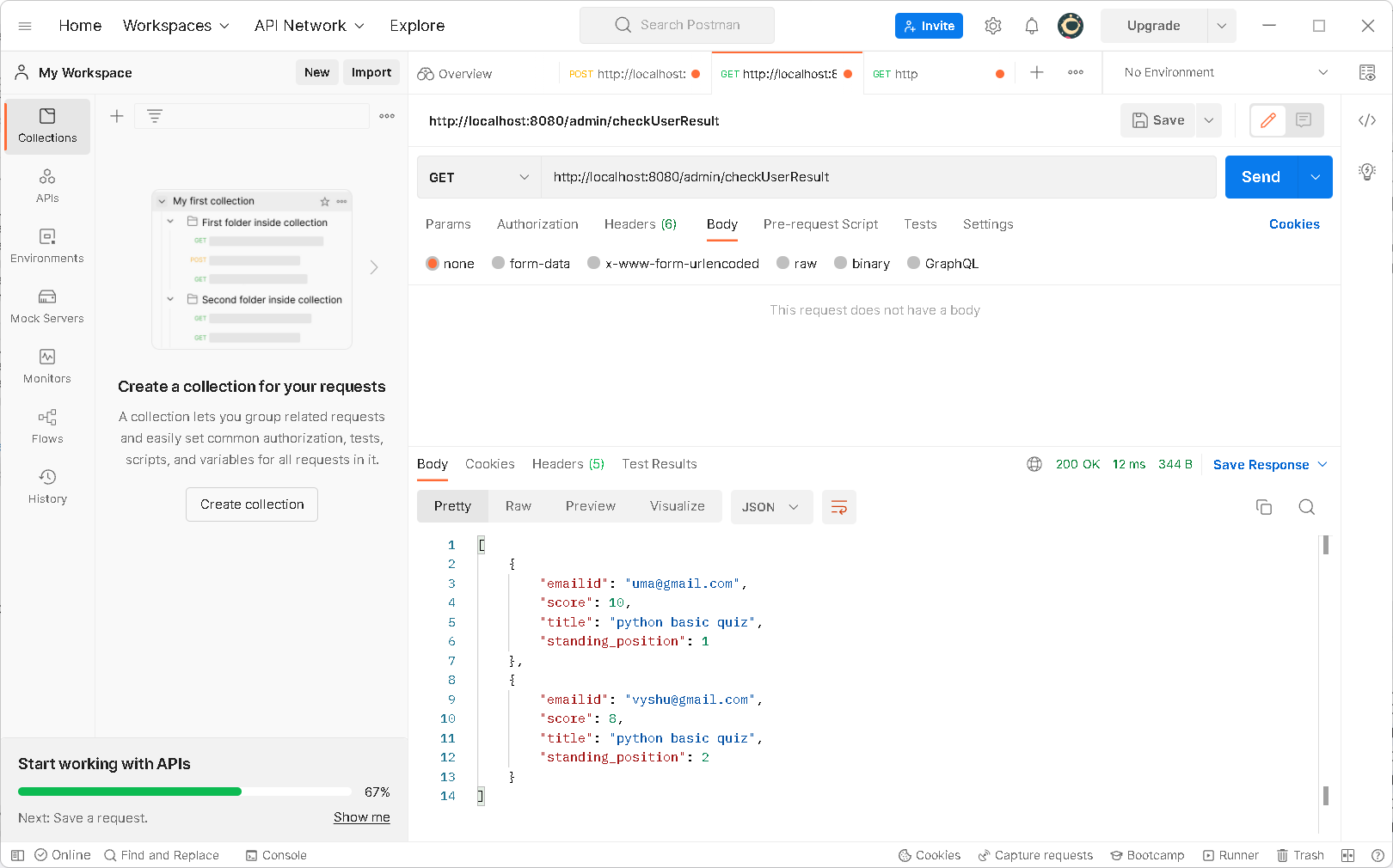
****

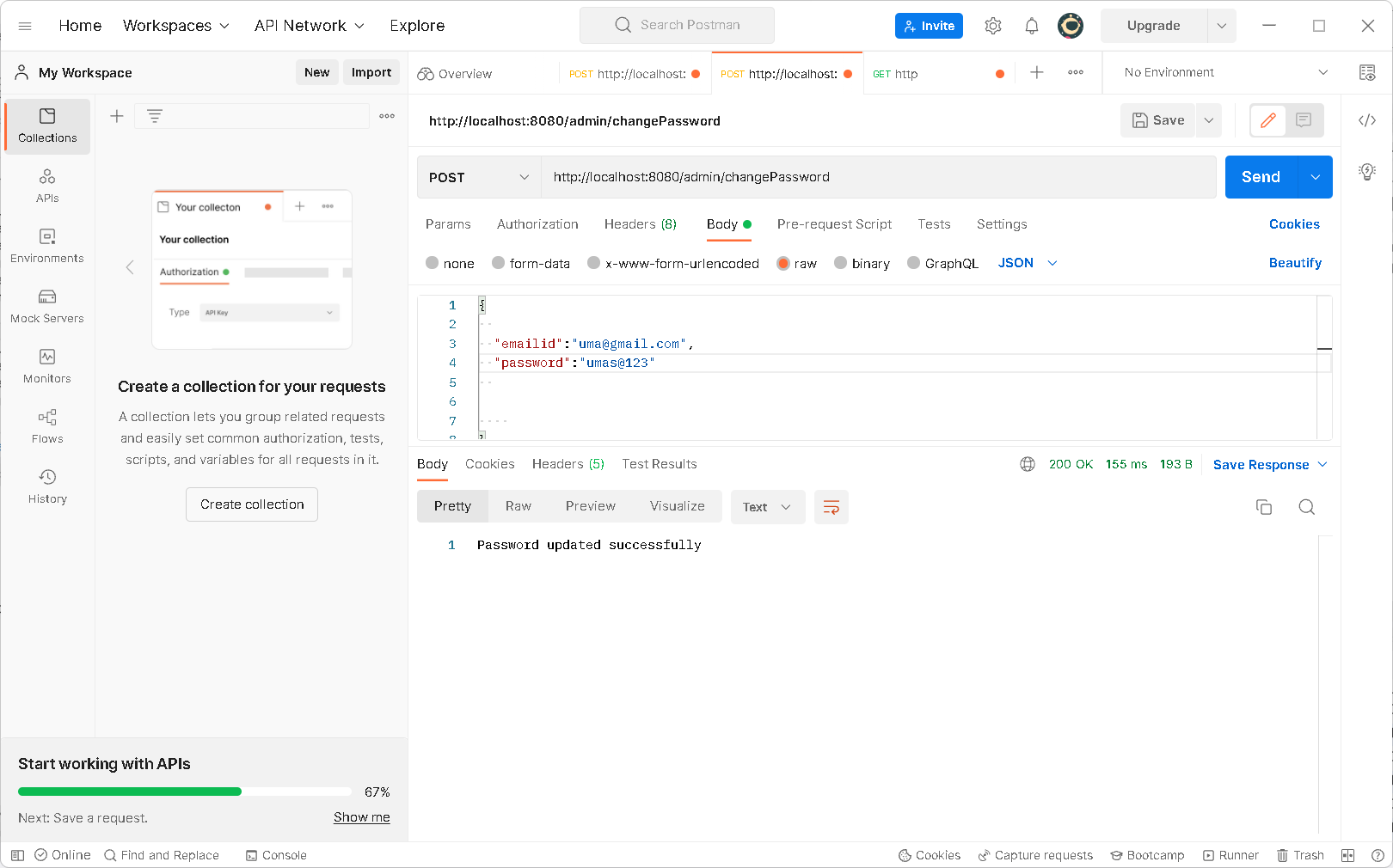
****

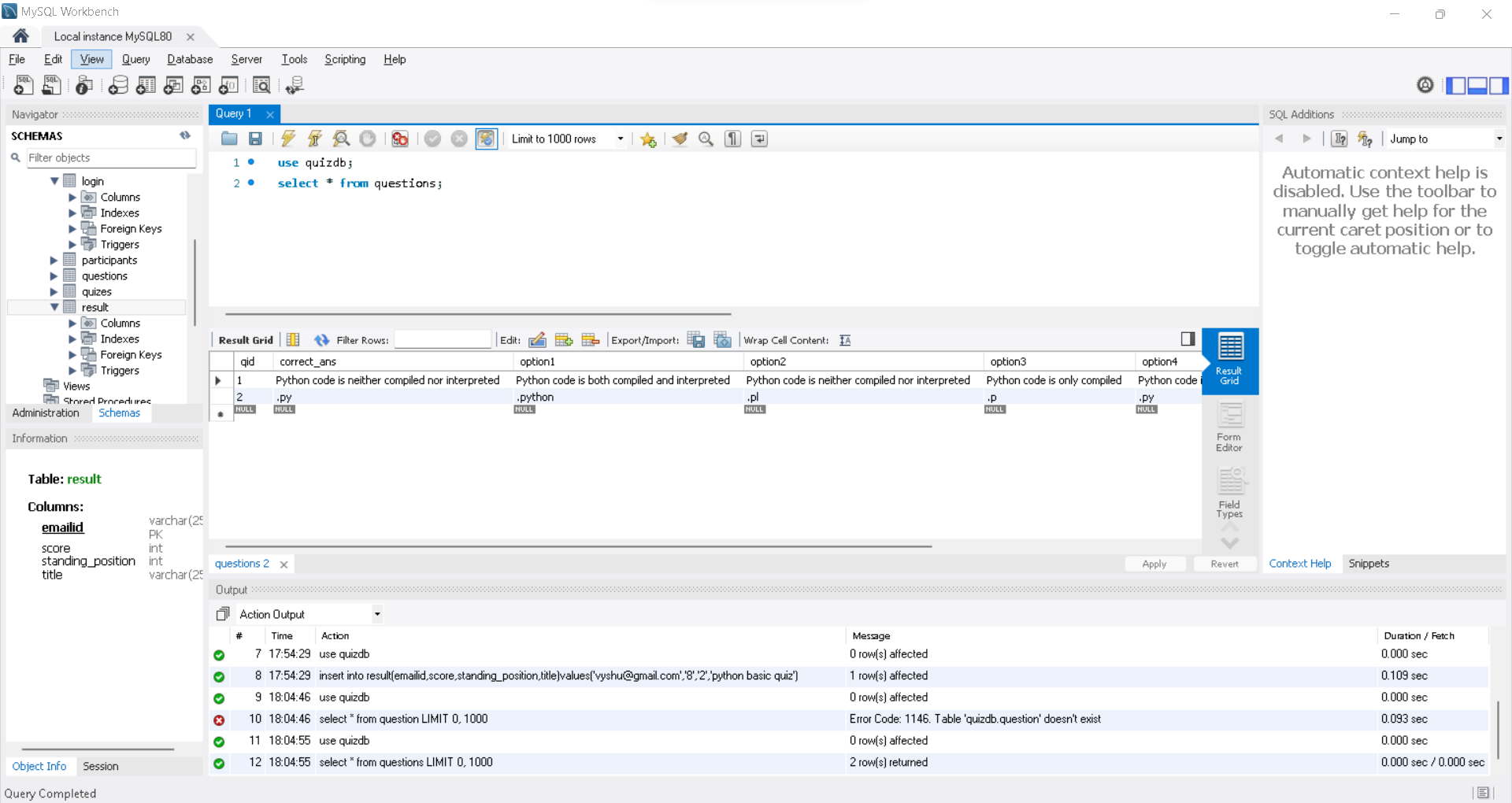
****

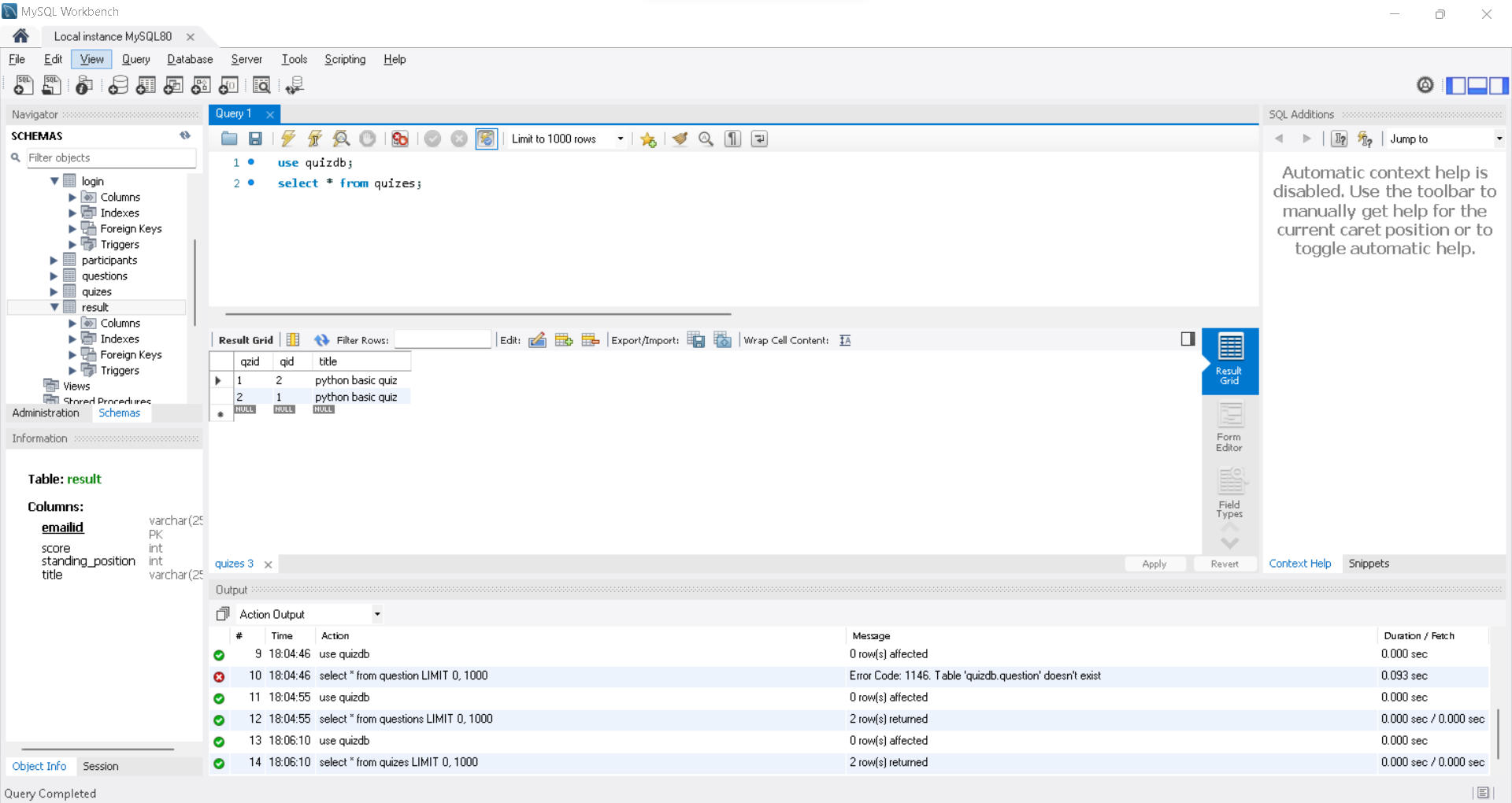
****

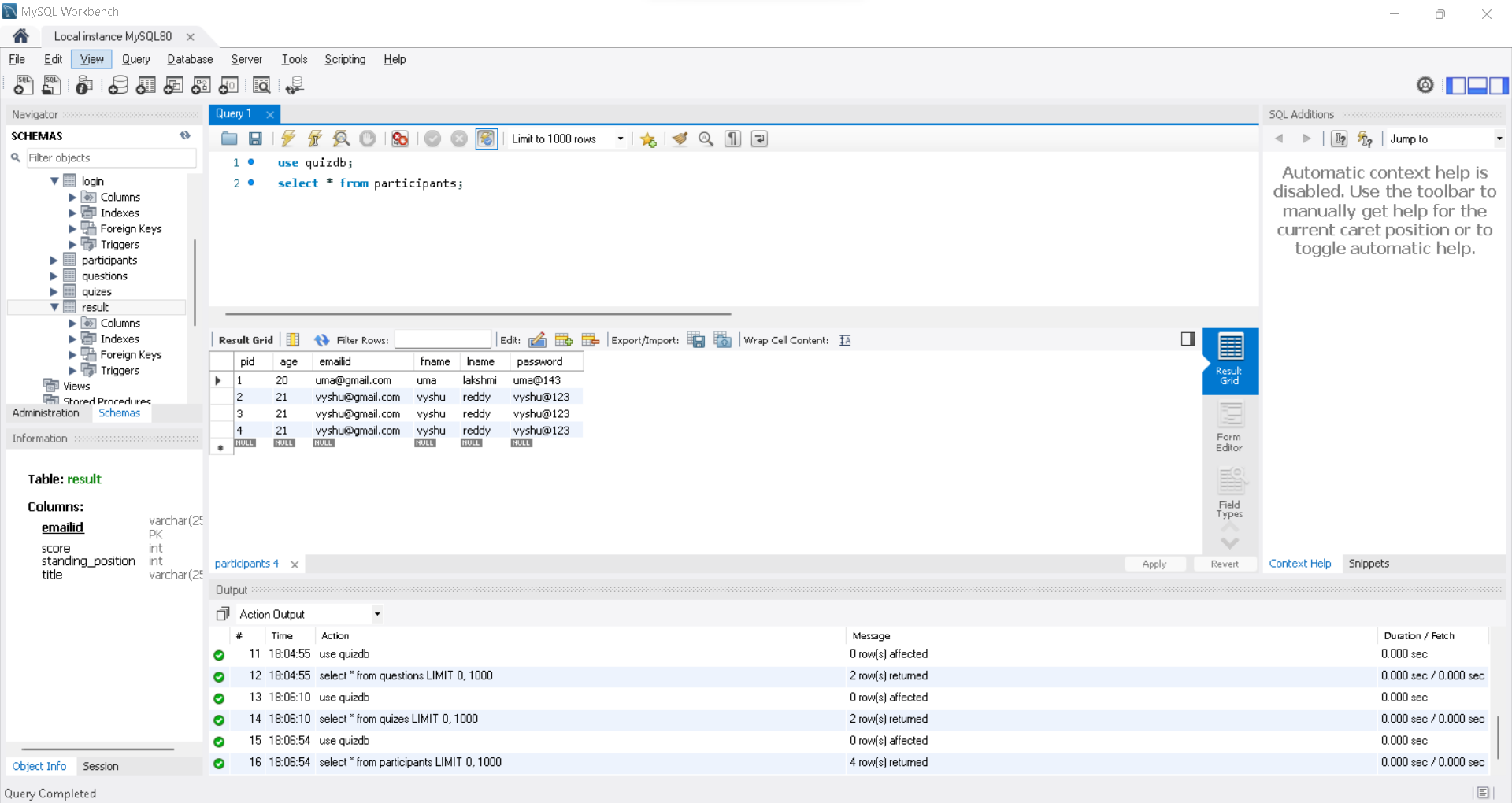
****

****

****

****

****

****