**My Education Results**

Completion table

|  |  |
| --- | --- |
| Lesson | Acknowledged with lesson & Task Done |
| English for Programmers | |
| 1. Basic vocabulary for IT | + |
| 1. Official emails | + |
| 1. Requirements analysis and SRS | + |
| 1. Development | + |
| 1. Testing | + |
| 1. Workplace challenges | + |
| 1. Troubleshooting | + |
| 1. Teamwork | + |
| 1. Negotiation tips | + |
| 1. Meetings & presentations | + |
| 1. English for online Conferences | + |
| Testing Basics & Terminology | |
| 1. Testing methods and types | + |
| 1. Testing instruments | + |
| 1. Testing artifacts | + |
| 1. Test design | + |
| 1. Client-server & file-server | + |
| 1. How to search information | + |
| 1. HTML CSS | + |
| 1. Browser developer tools | + |
| 1. Software development cycles | + |
| Git | |
| 1. Git & GitHub | + |
| 1. Work story & branches | + |
| 1. Team work | + |
| 1. Advanced features | + |
| Java Code Language | |
| 1. Introduction to Java | + |
| 1. Variables & operators | + |
| 1. Primitive data types, conditional operators, out of type boundaries and calculation accuracy | + |
| 1. Testability, auto tests, objects & methods | + |
| 1. Maven & Junit5 | + |
| 1. Cycles, parameter test & annotations | + |
| 1. Java auto-tests & CI | + |
| 1. Object-Oriented Programing (Basic) | + |
| 1. Object-Oriented Programing (Advanced) | + |
| 1. Objects with inner state | + |
| 1. Inheritance & scaling | + |
| 1. Exceptions | + |
| 1. Interfaces & generics | + |
| 1. Collection frameworks | + |
| Automation | |
| 1. Introduction to automation | + |
| 1. API testing | + |
| 1. Web interfaces testing | + |
| 1. Selenium & Selenide | + |
| 1. Patterns | + |
| 1. BDD | + |
| 1. Docker & Docker compose | + |
| 1. SQL | + |
| 1. Reporting & analysis | + |
| 1. Final lection | + |
| Quality Assurance diploma project | + |
| Corporate, HR & Work culture | |
| 1. How to find work | + |
| 1. How to write your resume | + |
| 1. How to pass the interview | + |
| 1. Net communication | + |
| 1. Corporate culture | + |
| My additional studies | |
| 1. C# & Azure.Net | + |
| 1. Javascript & jQuery | + |
| 1. Amazon Web Services | + |
| 1. Python | + |
| 1. Pipelines & CI | + |
| 1. Basic computer science theory | + |
| 1. Assembler | + |
| 1. Math formulas in programming | + |
| 1. Mind maps & algorithms in IT | + |
| 1. Game Engines & 3D graphics | + |
| 1. Android studio & Kotlin | + |
| 1. Testing metrics | + |
| My additional experience | |
| 1. Soft Skills Advanced | + |
| 1. S. Kulikov (from Epam systems) “Testing” Book | + |
| 1. Other Programming & Testing Books | + |
| 1. Html CSS | + |
| 1. SQL | + |
| 1. Game modding | + |
| 1. Windows admin skills | + |
| 1. Linux command line/admin | + |
| 1. Bugs finding / reporting | + |
| 1. IT links & Internet resources | + |
| 1. Alpha/Beta tests for games | + |
| 1. Android/IOS OS admin skills | + |
| 1. IT-hardware enthusiast (smartphones & HDD repairing, PC building, routers & commutators setup) | + |
| 1. IT-News tracking/community | + |
| 1. All study source material organized | + |
| 1. Patterns for work | + |
| 1. Resource & time management | + |
| 1. Software experience/clouds | + |
| 1. Office Word, Excel, Power Point | + |
| 1. Visio, Xmind, Google Graphs, MS Project | + |
| 1. Tracking new code technologies | + |
| 1. OC Install on PC, iPhone, Mac, Linux, Android. etc. | + |
| 1. Macros writing | + |
| Testing in Visual Studio (C/C++/C#) | |
| 1. Testing tools in Visual Studio | + |
| 2. Unit test tools and tasks | + |
| 3. IntelliTest | + |
| 4. Test Explorer | + |
| 5. Write unit tests for managed code | + |
| 6. Microsoft Fakes | + |
| 7. Unit tests for C/C++ code | + |
| 8. Write unit tests for C/C++ DLLs | + |
| 9. Code coverage | + |
| 10. Live Unit Testing | + |
| 11. Web performance and load testing | + |
| 12. Test mix | + |
| 13. Counter sets | + |
| 14. Load test | + |
| 15. Test controllers and test agents | + |
| 16.API Tests | + |
| 17. Collect diagnostics & Test Results | + |
| 18. UI automation using Coded UI test | + |
| 19. Remote testing in Visual Studio | + |
| Postman | |
| 1. Getting Started | + |
| 2. Sending Requests | + |
| 3. Writing Scripts | + |
| 4. Running Collections | + |
| 5. Collaborating in Postman | + |
| 6. Designing and Developing your API | + |
| 7. Publishing & Monitoring your API | + |
| 8. API security | + |
| 9. Administration | + |
| 10. Developer Resources & Integrations | + |
| 11.Bootcamp | + |
| Appium | |
| 1.Getting Started | + |
| 2.Writing & Running Tests | + |
| 3.Commands | + |
| 4.Advanced | + |
| 5.Documentation | + |
| 6.Devs Overview | + |
| Selenium & Selenide | |
| 1. WebDriver | + |
| 2.Capabilities | + |
| 3.Browser | + |
| 4.Elements | + |
| 5.Remote WebDriver | + |
| 6.Drivers | + |
| 7.Waits | + |
| 8.Actions API | + |
| 9.Grid | + |
| 10.Test Practices | + |

Plan to improve my skills and profession:

1. ISQB – certificate
2. English practice (verbal)
3. JavaScript (advanced)
4. Time management