This course supports the assessments for DJV1. The course covers 6 competencies and represents 4 competency units.

**Introduction**

**Overview**
This course will provide you with the core fundamentals for software development. You will learn about the tools, technologies, and methodologies used for developing working software systems. You will study how to develop algorithms and use programming structures to create working programs. The fundamentals of Object-Oriented programming and databases will be covered, as well as the basics of web-based and desktop applications. Upon completion of this course, you will have a solid understanding of how to design, develop, and test software applications.

Certification exams attempt to measure test-takers' skills rather than their knowledge of a subject, such as a traditional school's tests would measure. This is particularly true for scripting and programming exams since these are active disciplines. Accordingly, this course has an emphasis on hands-on exercises. Make sure to complete all exercises during you preparation for the assessment. If you lack the skill to complete them, review the corresponding chapter and try again.

Watch the following introduction video for this course:

**Competencies**
This course provides guidance to help you demonstrate the following 6 competencies:

- **Competency 400.1.1: Fundamentals of Core Programming**
  The graduate analyzes core elements and tools used in the design and trace of computer programs.
- **Competency 400.1.2: Fundamentals of Object-Oriented Programming**
  The graduate applies the key principles of object-oriented programming.
- **Competency 400.1.3: Fundamentals of Software Development**
  The graduate applies lifecycle management activities to the development and testing of software applications.
- **Competency 400.1.4: Fundamentals of Web Applications**
  The graduate applies core elements and tools for developing Web applications.
- **Competency 400.1.5: Fundamentals of Desktop Applications and User Interfaces**
  The graduate applies fundamental principles, concepts, and tools used in the development of desktop applications and user interfaces.
- **Competency 400.1.6: Fundamentals of Databases**
  The graduate applies fundamental principles, concepts, and tools used in the development and management of relational databases.

**Course Instructor Assistance**
As you prepare to successfully demonstrate competency in this subject, remember that course instructors stand ready to help you reach your educational goals. As subject matter experts, mentors enjoy and take pride in helping students become reflective learners, problem solvers, and critical thinkers. Course instructors are excited to hear from you and eager to work with you.

Successful students report that working with a course instructor is the key to their success. Course instructors are able to share tips on approaches, tools, and skills that can help you apply the content you're studying. They also provide guidance in assessment preparation strategies and troubleshoot areas of deficiency. Even if things don’t work out on your first try, course instructors act as a support system to guide you through the revision process. You should expect to work with course instructors for the duration of your coursework, so you are welcome to contact them as soon as you begin. Course instructors are fully committed to your success!

Preparing for Success

The information in this section is provided to detail the resources available for you to use as you complete this course.

Learning Resources

The learning resources listed in this section are required to complete the activities in this course. For many resources, WGU has provided automatic access through the course. However, you may need to manually enroll in or independently acquire other resources. Read the full instructions provided to ensure that you have access to all of your resources in a timely manner.

Automatically Enrolled Learning Resources

You will be automatically enrolled at the activity level for the following learning resources. Simply click on the links provided in the activities to access the learning materials.

VitalSource E-Text

The following textbook is available to you as an e-text within this course. You will be directly linked to the specific readings required within the activities that follow.


The *98-361: MTA Software Development Fundamentals* textbook covers all the objectives covered on Software Development Fundamentals, exam 98-361. Before starting a lesson, review the "Lesson Skill Matrix" to get a sense of topics covered by the lesson. As you proceed through the lesson material, be sure to complete each practice exercise. These are easily identified by the picture of an arrow with a "GET READY" cue. Each lesson has several hands-on practice exercises, giving you a chance to apply the programming concepts. When learning any programming topic, practice is paramount. You need the practice in order to be adequately prepared for the exam. Each lesson also points out key certification topics to which you should pay close attention.

*Note: This e-text is available to you as part of your program tuition and fees, but you may*
*purchase hard copies at your own expense through a retailer of your choice. If you choose to do so, please use the ISBN listed to ensure that you receive the correct edition.*

**MeasureUp**
You will access the following resource from MeasureUp at the activity level within this course:

- MTA: Software Developer Fundamentals

Please use the steps listed in following document to guide your study of the MeasureUp material:

- [Using MeasureUp](#)

**LearnKey**
You will use the following learning resource from LearnKey at the end of your course to review any topics as needed:

- MTA Software Development Fundamentals

**IMPORTANT:** There are some important items to understand regarding the use of the LearnKey resource, primarily that it functions best with the Firefox browser. It is recommended that you use Firefox as you utilize the LearnKey resource. If you have technical assistance needs regarding this resource, you are invited to use LearnKey’s [Tech Support Facts](#).

**Other Learning Resources**
You will use the following learning resources for this course.

**MS Imagine**
Access the following instructions for obtaining free Microsoft software:

- [MS Imagine](#)

**Microsoft Visual Studio Professional**
To practice writing code for the exercises presented in each lesson, you will use an integrated development environment. Microsoft Visual Studio for Windows Desktop is easy to use and was designed for beginning programmers. You can access Microsoft Visual Studio through your Ms Imagine account. If you have already downloaded Microsoft Visual Studio from a previous course, you do not need to reinstall.

**Activate the MTA Practice Test (Optional)**

The text includes access to a practice test product. The MTA Practice Test is an optional resource. The procedure for activating the test and the required code are listed in the front matter. Please note the technical support contact email on the same page if you have questions about installation or code activation.

**Pacing Guide**
The pacing guide suggests a weekly structure to pace your completion of learning activities. It is provided as a suggestion and does not represent a mandatory schedule. Follow the pacing guide carefully to complete the course in the suggested timeframe.

- Pacing Guide: Software Development Fundamentals

Note: This pacing guide does not replace the course. Please continue to refer to the course for a comprehensive list of the resources and activities.

Microsoft Official Academic Course: Text

There are six lessons in your textbook. As you read through each lesson, you will be provided an opportunity to apply the software development skills encountered. Please take the time to complete these "Get Ready" practice exercises. This is a critical step in developing your programming competencies. To learn how to write programs, you must practice. The more you code, the more proficient you will become. These lessons cover all exam objectives and should be completed in order.

You will begin by learning how to develop algorithms and writing programs. You will learn about the activities associated with each phase of the software development lifecycle and get exposure to writing web applications and desktop applications. Because some applications interact with a database, the basics of databases are included. The fundamentals of object oriented programming are also covered, giving you an understanding of class design, inheritance, polymorphism, and encapsulation. All of these topics will prepare you for the upcoming programming coursework you will complete in your degree program.

Lesson 1: Introduction to Programming
This lesson introduces you to the fundamentals of programming. You will learn about algorithm development and using decision structures and loops.

This topic addresses the following competencies:

- Competency 400.1.1: Fundamentals of Core Programming
  The graduate analyzes core elements and tools used in the design and trace of computer programs.
- Competency 400.1.2: Fundamentals of Object-Oriented Programming
  The graduate applies the key principles of object-oriented programming.
- Competency 400.1.3: Fundamentals of Software Development
  The graduate applies lifecycle management activities to the development and testing of software applications.
- Competency 400.1.4: Fundamentals of Web Applications
  The graduate applies core elements and tools for developing Web applications.
- Competency 400.1.5: Fundamentals of Desktop Applications and User Interfaces
  The graduate applies fundamental principles, concepts, and tools used in the development of desktop applications and user interfaces.
- Competency 400.1.6: Fundamentals of Databases
  The graduate applies fundamental principles, concepts, and tools used in the
development and management of relational databases.

**Reading: Lesson 1**

Read the following in the *98-361: MTA Software Development Fundamentals* textbook:

- lesson 1 ("Introduction to Programming")

Pay close attention to the techniques and tools provided for algorithm development. Algorithms are your logic. They will be used to design the processing that takes place in your applications. Using flowcharts and pseudo code to outline your logic are necessary to developing working applications. This topic is your foundation and is critical to your success as a programmer.

You will need to practice writing code by using the latest version of Microsoft Visual Studio Express or Microsoft Visual Studio. Be sure that you have installed either program at this time. If you have difficulty with any of the coding practice exercises, please schedule time with the course instructor to get help.

*Notice that the exercise labeled "Use The Recursive Method" located in page 23 of the textbook produces no visible output even though the code syntax is correct and the program runs successfully. If you want to see the output, replace the line of code on step 2 with the following:* `Console.WriteLine(Factorial(5));`

**Lesson 1 Knowledge Assessment**

Take the knowledge assessment at the end of lesson 1 ("Introduction to Programming").

Do not continue until you have answered everything correctly, and understand what you missed. Contact the course instructor, if needed. The competency and proficiency assessments are optional.

**Lesson 2: Introduction to Object-Oriented Programming**

You will learn the fundamentals of using objects and classes in this lesson. All of the fundamentals covered in this lesson are transferable to any object-oriented programming language.

This topic addresses the following competencies:

- Competency 400.1.1: Fundamentals of Core Programming
  The graduate analyzes core elements and tools used in the design and trace of computer programs.
- Competency 400.1.2: Fundamentals of Object-Oriented Programming
  The graduate applies the key principles of object-oriented programming.
- Competency 400.1.3: Fundamentals of Software Development
  The graduate applies lifecycle management activities to the development and testing of software applications.
- Competency 400.1.4: Fundamentals of Web Applications
  The graduate applies core elements and tools for developing Web applications.
• Competency 400.1.5: Fundamentals of Desktop Applications and User Interfaces
  The graduate applies fundamental principles, concepts, and tools used in the development of desktop applications and user interfaces.
• Competency 400.1.6: Fundamentals of Databases
  The graduate applies fundamental principles, concepts, and tools used in the development and management of relational databases.

Reading: Lesson 2

Read the following in the *98-361: MTA Software Development Fundamentals* textbook:

  • lesson 2 ("Introduction to Object-Oriented Programming")

The concepts covered in this lesson will make sense if you complete each practice exercise. Do not skip any hands-on practice exercises. You will be introduced to the principles of encapsulation, inheritance, and polymorphism, which are at the heart of object-oriented programming.

**Lesson 2: Knowledge Assessment**

Take the knowledge assessment at the end of lesson 2 ("Introduction to Object-Oriented Programming").

Do not continue until you have answered everything correctly and understand what you missed. Contact the course instructor, if needed. The competency and proficiency assessments are optional.

**Lesson 3: Understanding General Software Development**

Successful developers need to have an understanding of all activities involved with developing a software system. You will learn about life cycle management.

This topic addresses the following competencies:

• Competency 400.1.1: Fundamentals of Core Programming
  The graduate analyzes core elements and tools used in the design and trace of computer programs.
• Competency 400.1.2: Fundamentals of Object-Oriented Programming
  The graduate applies the key principles of object-oriented programming.
• Competency 400.1.3: Fundamentals of Software Development
  The graduate applies lifecycle management activities to the development and testing of software applications.
• Competency 400.1.4: Fundamentals of Web Applications
  The graduate applies core elements and tools for developing Web applications.
• Competency 400.1.5: Fundamentals of Desktop Applications and User Interfaces
  The graduate applies fundamental principles, concepts, and tools used in the development of desktop applications and user interfaces.
• Competency 400.1.6: Fundamentals of Databases
  The graduate applies fundamental principles, concepts, and tools used in the
development and management of relational databases.

**Reading: Lesson 3**

Read the following from the 98-361: MTA Software Development Fundamentals textbook:

- lesson 3 ("Understanding General Software Development")

When writing software systems, you are concerned with behavior and data. Through the reading and practice exercises, you will learn how to utilize appropriate data structures to build a software solution. You will take a close look at activities involved with determining requirements, designing systems, and the various types of testing that can be applied. This will give you the big picture of software lifecycle management.

**Lesson 3 Knowledge Assessment**

Take the knowledge assessment at the end of lesson 3 ("Understanding General Software Development").

Do not continue until you have answered everything correctly, and understand what you missed. Contact the course instructor, if needed. The competency and proficiency assessments are optional.

**Lesson 4: Understanding Web Applications**

This lesson covers all of the basics of website, web application and web services development.

This topic addresses the following competencies:

- **Competency 400.1.1: Fundamentals of Core Programming**
  The graduate analyzes core elements and tools used in the design and trace of computer programs.

- **Competency 400.1.2: Fundamentals of Object-Oriented Programming**
  The graduate applies the key principles of object-oriented programming.

- **Competency 400.1.3: Fundamentals of Software Development**
  The graduate applies lifecycle management activities to the development and testing of software applications.

- **Competency 400.1.4: Fundamentals of Web Applications**
  The graduate applies core elements and tools for developing Web applications.

- **Competency 400.1.5: Fundamentals of Desktop Applications and User Interfaces**
  The graduate applies fundamental principles, concepts, and tools used in the development of desktop applications and user interfaces.

- **Competency 400.1.6: Fundamentals of Databases**
  The graduate applies fundamental principles, concepts, and tools used in the development and management of relational databases.

**Reading: Lesson 4**

Read the following from the 98-361: MTA Software Development Fundamentals textbook:
The reading and practice exercises will give you an opportunity to work with HTML, CSS, and JavaScript. This may be material you are already familiar with from a previous course. If that is the case, you may shorten the amount of time you spend by just focusing on completing the practice exercises. Be sure you do not skip the section on state management or web services development.

**Lesson 4 Knowledge Assessment**

Take the knowledge assessment at the end of lesson 4 ("Understanding Web Applications").

Do not continue until you have answered everything correctly, and understand what you missed. Contact the course instructor, if needed. The competency and proficiency assessments are optional.

**Lesson 5: Understanding Desktop Applications**

You may need to develop a smart client application that runs without being connected to the Internet. This lesson covers how to create a visually appealing Windows Form application.

This topic addresses the following competencies:

- **Competency 400.1.1: Fundamentals of Core Programming**
  The graduate analyzes core elements and tools used in the design and trace of computer programs.
- **Competency 400.1.2: Fundamentals of Object-Oriented Programming**
  The graduate applies the key principles of object-oriented programming.
- **Competency 400.1.3: Fundamentals of Software Development**
  The graduate applies lifecycle management activities to the development and testing of software applications.
- **Competency 400.1.4: Fundamentals of Web Applications**
  The graduate applies core elements and tools for developing Web applications.
- **Competency 400.1.5: Fundamentals of Desktop Applications and User Interfaces**
  The graduate applies fundamental principles, concepts, and tools used in the development of desktop applications and user interfaces.
- **Competency 400.1.6: Fundamentals of Databases**
  The graduate applies fundamental principles, concepts, and tools used in the development and management of relational databases.

**Reading: Lesson 5**

Read the following from the 98-361: MTA Software Development Fundamentals textbook:

- **lesson 5 ("Understanding Desktop Applications")**

The reading and exercises in this lesson will introduce you to Visual Studio’s Windows Forms designer. The activities may seem Windows-centric, but there are other integrated development environments that have similar tools, such as NetBeans and Eclipse. The knowledge learned
here is easily transferred to other tools for building desktop applications. These activities will also help develop your understanding of event-based programming.

**Lesson 5 Knowledge Assessment**

Take the knowledge assessment at the end of lesson 5 ("Understanding Desktop Applications").

Do not continue until you have answered everything correctly, and understand what you missed. Contact the course instructor, if needed. The competency and proficiency assessments are optional.

**Lesson 6: Understanding Databases**

Software developers have to code behavior and work with data. To make systems dynamic, being able to connect to a database and write an application that interacts with a database is a critical industry need.

This topic addresses the following competencies:

- **Competency 400.1.1: Fundamentals of Core Programming**
  The graduate analyzes core elements and tools used in the design and trace of computer programs.

- **Competency 400.1.2: Fundamentals of Object-Oriented Programming**
  The graduate applies the key principles of object-oriented programming.

- **Competency 400.1.3: Fundamentals of Software Development**
  The graduate applies lifecycle management activities to the development and testing of software applications.

- **Competency 400.1.4: Fundamentals of Web Applications**
  The graduate applies core elements and tools for developing Web applications.

- **Competency 400.1.5: Fundamentals of Desktop Applications and User Interfaces**
  The graduate applies fundamental principles, concepts, and tools used in the development of desktop applications and user interfaces.

- **Competency 400.1.6: Fundamentals of Databases**
  The graduate applies fundamental principles, concepts, and tools used in the development and management of relational databases.

**Reading: Lesson 6**

Read the following from the 98-361: MTA Software Development Fundamentals textbook:

- lesson 6 ("Understanding Databases")

This lesson will arm you with exposure to tools and techniques that will help you develop applications that query and update a database. It remains important to complete all hands-on practice exercises in the lesson.

**Lesson 6 Knowledge Assessment**

Take the knowledge assessment at the end of lesson 6 ("Understanding Databases").
Do not continue until you have answered everything correctly, and understand what you missed. Contact the course instructor, if needed. The competency and proficiency assessments are optional.

**MeasureUp and LearnKey**

The MeasureUp and LearnKey learning resources are meant to complement the other learning resources for this course.

**MeasureUp Software and LearnKey**

The MeasureUp testing software and LearnKey course provide you with the necessary tools to reinforce your learning and validate your knowledge for certification.

---

**MeasureUp Testing**

Complete the following MeasureUp material:

- **MTA: Software Developer Fundamentals**

Please use the steps listed in following document to guide your study of the MeasureUp material:

- **Using MeasureUp**

**LearnKey**

Use the following LearnKey resource to review the course content and study the specific items necessary to complete your competency:
• **MTA Software Development Fundamentals**

**IMPORTANT:** There are some important items to understand regarding the use of the LearnKey resource, primarily that it functions best with the Firefox browser. It is recommended that you use Firefox as you utilize the LearnKey resource. If you have technical assistance needs regarding this resource, you are invited to use LearnKey’s [Tech Support Facts](#).

**Final Steps**

Congratulations on completing the activities in this course! This course has prepared you to complete the assessments associated with this course. If you have not already been directed to complete the assessments, schedule and complete your assessments now.

**First Attempt Checklist**

One of the many things that makes WGU unique is its competency based education model. If you know the material, all you have to do is prove it by passing the exam. If you can do this, you can accelerate the receipt of your degree.

To make sure you have the best chance possible to pass the exam on your first attempt, the following should be completed before you refer to take it:

2. Complete MeasureUp in Certification Mode and score at least 80% or better on the exam at least two times before attempting the final exam.

If this has been done, and you feel comfortable with all of the concepts presented, you are most likely ready to refer for the first attempt of the exam.

If the first attempt is failed, you will be required to get in contact with the course instructor to see what went wrong and how you can prepare to make sure your second attempt is a success. Once the course instructor feels you are ready to refer for the second attempt of the exam, he or she will approve your second exam referral.

**Outside Vendor Assessment**

Complete the Microsoft MTA Software Development Fundamentals Exam 98-361:

- For details on the exam, please visit the [Exam 98-361](#) information page.

**Software Development Fundamentals Exam Referral Instructions**

1. **Create a Certiport Account**
   In order to refer for your final exam, you will first need to create an account with [Certiport](#). Click ‘Register’ and follow the required steps. Please keep your login information handy—you will need to log into your Certiport account to take your assessment.

2. **Choose an Assessment Location**
   The assessment has to be taken at a proctored site; there are two options for choosing a site:
a. The Certiport website has a network of testing sites from which to choose.

b. If you cannot find a Certiport site that is close to you, you can search for a site in the "WGU Proctor Yellow Pages." This site would have to be set up by Certiport, please follow step 4.2

3. Contact the Certiport Testing Center to Schedule Your Exam

a. Most testing centers listed on the Certiport locator will provide a phone number for you to call and schedule, others will list an e-mail address or have a link to their website. Please use their preferred method to arrange your assessment.

Please note: Some testing centers require a proctor fee that has to be paid to the site, independent from the voucher number that pays for the exam itself. Check with the site on the amount of the proctor fee and how it can be paid to them.

Once you have a scheduled test date and time, follow step 4.

b. If the testing center will not allow you to make an appointment, please proceed to step 4.3

4. Submit Your Referral

4.1 To complete the scheduling process for an exam on your Degree Plan, click the ‘schedule now’ button for this assessment. You will be required to provide the date and time your exam is scheduled, the name of the testing center, and any proctor fees; please make sure you have this information available.

4.2 If you cannot find a site or found a site in the WGU network that you would like to become a Certiport testing center, click the checkbox next to ‘I need help finding a site’ and enter the name of the WGU site or the reason why you need help scheduling in the notes section underneath. This will cancel the requirement of entering any of the other required information the referral is asking for and will allow you to submit the referral. A scheduler will then work with you to get a site set up for you.

4.3 If the Certiport testing site does not allow you to make an appointment without a voucher number for this assessment, click the checkbox next to ‘I need help finding a site’ and enter in the notes section underneath which site you selected and that they require a voucher number to schedule your appointment.

IMPORTANT: WGU will provide a discounted Certiport voucher for this assessment and will pay all associated proctor fees. You will receive the voucher within 3 business days of submitting the referral. Please understand that you will only be reimbursed the amount WGU would pay for a discounted voucher if you purchased a voucher directly from Certiport. However, you will be reimbursed the full amount of any proctor fee the site might charge you if WGU cannot pay the site directly. Submit receipts to reimbursements@wgu.edu along with your current mailing address and the date and exam code and you will receive a check within 3 weeks.
5. Verify Your Information
Make sure you fill out the information as requested before submitting the referral. Double check to make sure everything is correct and in order. Please provide any other information you feel your scheduler needs to know before processing this request for you.

Submitting External Assessment Scores

Your score report will be submitted by Certiport within 5 days of your exam and added to your account by the records department. If your account does not reflect the changes in this time frame, make your mentor aware of the situation.