Junior Programmer Pathway

Teacher preparation guide

What is the Junior Programmer Pathway?

The Junior Programmer Pathway is designed for anyone interested in learning to code or obtaining an entry-level Unity role. This pathway assumes a basic knowledge of Unity and has no math prerequisites. Junior Programmer prepares you to get Unity Certified so that you can demonstrate your job readiness to employers.

Key details

A 12 to 14-week learning journey that teaches programming in Unity, and is designed for anyone who wants to become familiar with the process of creating C# scripts.

The Junior Programmer Pathway covers all the basic concepts and skills to introduce you to C# in Unity, and get you started on the path to becoming a Unity developer.

Scope of this document

This teacher preparation guide accompanies the Junior Programmer Pathway and will help you get ready to bring this curriculum to your classroom.

Table of contents

Teacher orientation

1a. Download/print the teacher training worksheet and understand how to use it		
Purpose of worksheet	→ Help track your progress through teacher training → Help plan and customize the course curriculum for the classroom	□

1b. Understand the course objectives, requirements, and structure		
Course objectives	→ C# skills → Unity skills → Project management skills	□
Course requirements	→ This pathway assumes a basic knowledge of Unity → Mac or PC with standard mouse required (* headphones recommended) → 12 weeks minimum	□
Course structure	→ Prototypes | Lessons → Assessments | Challenges & Quizzes → Personal Projects | Labs → Relationship between prototypes, assessments, and personal projects	□

1c. Familiarize yourself with the course content and available resources

Junior Programmer Pathway	→ Website: https://learn.unity.com/pathway/junior-programmer → Mapping between online course and Syllabus / Scope & Sequence → Online course navigation, including "For Educators" tab	□
Lessons: online vs in-class	→ Where to find lesson plans → Components of a lesson (overview, introduction, steps, context, instructions, screenshot / code snippets, recaps) → Mapping between lesson plans and online lessons → How a lesson could be teacher-led in a classroom → Importance of "Watch, then Do" for independent or teacher-led instruction	□
Challenges	→ How challenges work	□
Quizzes	→ How quizzes work	□
Labs: independent or in-groups	→ How labs/personal projects are different than lessons/prototypes → How labs could be completed at home or in-class → How labs could be completed independently or in groups	□
Bonus Features	→ How bonus features work	□

1d. Understand who your learners are

The Unity Junior Programmer Pathway is a comprehensive entry point for getting started with C# development in Unity, specifically designed for those with no prior experience. Depending on the profile and prior experience of your learners, you can use it to facilitate a range of different experiences to best meet their needs.

Learner age range	**Delivery suggestion**
Lower secondary (middle school and junior high)	→ Structured, facilitated sessions throughout, that break down the self-paced technical instructions into sessions with extension opportunities to ensure the group keeps pace → Scaffolding and extension options mapped to those sessions will help provide differentiated learning experiences → The software installation/new user

	onboarding guidance is unlikely to be required for this age range
Upper secondary (high school)	→ Independent completion of the self-paced technical learning content, with scaffolding and extension options to provide differentiated learning experiences → Facilitated research and discussion sessions on creator skills and real-time industry exploration → The software installation/new user onboarding guidance is unlikely to be required for this age range

Design your education experience

2a. Adapting Junior Programmer content for different teaching approaches and contexts

As an instructor/facilitator for a learning experience based around Junior Programmer, your most valuable contributions are likely to be:

- ➔ Basic scripting techniques and paradigms of C# development in Unity (this is especially the case for less technically literate cohorts)
- ➔ Facilitating discussion and exploration of creator skills and workplace industries
- ➔ Questioning to consolidate and deepen understanding
- ➔ Troubleshooting participant technical issues.

The following table offers some guidance on adapting this learning experience for your teaching approaches and circumstances:

Flipped Classroom / instruction	Pre-class work can be assigned by tutorial or Mission within the Junior Programmer Pathway. Research tasks for creator skills and real-time industry group discussions, presentations, or peer review feedback sessions are also ideal for the flipped classroom.
Project-based	The Junior Programmer Pathway is broken into Missions, with each Mission containing smaller projects. This can be used for project-based learning.
Inquiry-based	The Junior Programmer Pathway covers basic software

	development fundamentals, and so has not been designed with inquiry-based learning as a priority. However, the career and real-time industry information within the Pathway could provide the foundation for identifying research questions for further inquiry-based/research-based learning that meets the particular needs of your group.
Careers and industry focus	There are no dependencies between the real-time industry content in the Junior Programmer Pathway and the technical tutorials. These can be adapted as best meets the needs of your class or integrated into a wider career-focused learning experience.

2b. Review common Pathway Configurations

	Lessons	**Challenges, quizzes, & bonus features**	**Personal Projects**	**% teacher - led % in-class**	**Relevant affordances and constraints**
1: Teacher-led	Teacher-led In-class	Independent In-class	Teacher-led In-class	80% teacher-led 100% in-class	Students can't work at home You want complete control You feel confident w/ material or can spend time on training
2. Teacher-augmented	Video-led In-class	Independent In-class	Video-led In-Class	0% teacher-led 100% in-class	Students can't work at home You do not feel confident with material yet and/or do not have time for training

2c. Determine your unique classroom affordances and constraints

Available hardware?	● In your classroom, do you have a way of projecting or displaying your own computer's screen so that the entire class can see it? *("Yes" allows for **teacher-led in-class** or **video-led in-class** activities.)*	**Yes | No**

	● Can a set of headphones be included at each computer station? *("Yes" allows for **independent in-class** activities.)*	**Yes | No**
Student work at-home?	● Can all of your students access a computer that can run Unity outside of class time? This could include getting access to the computer lab outside of their normal class period. *("Yes" allows for **at-home** activities.)*	**Yes | No**
Experience with material?	● *Do you either (a) have ~20 hours to dedicate to training and learning the material before the course begins or (b) already have a lot of experience teaching Unity and C#?* *(If "yes," **teacher-led** activities are an option for you. If "no," **independent** or **video-led** activities may be best to start.)*	**Yes | No**

2d. Choose a Pathway configuration that works for your classroom

Based on the affordances and constraints of your particular classroom (selected above), choose the configuration of each activity that best suits your needs.

Activity	**Choose your option:**	**Configuration Options**
Lessons →	______________________________	Teacher-led In-class
Challenges →	______________________________	Video-led In-class
Quizzes →	______________________________	In groups, In-class
Labs →	______________________________	Independent, In-class
Bonus Features →	______________________________	Independent, At-home

2e. Determine how much of the Pathway you should aim to complete

Determine if you can finish the entire Pathway.	How many combined in-class hours and at-home hours (if any) will the students have to work on this course?	____ ***hours***

	The entire pathway takes **12 weeks** to complete independently, but can take longer in a classroom depending on class size, experience, amount of time given to work on personal projects, and other factors. How long do you think it would take for your class to complete the course?	____ ***hours***
Units or activities to exclude from the curriculum (if any)	If the number of hours available is less than the number of hours required to complete the Pathway, you will have to exclude certain content. You can: a. exclude **entire units** (e.g. only do Units 1, 2, and 3), b. exclude **certain activity types** (e.g. do not do challenges or labs) c. exclude entire units ***and*** certain activity types (e.g. only do Units 1-3, not including labs)	***Exclude:*** unit 2/3/4/5 ***and/or Exclude:*** Challenges Quizzes Labs

Getting started checklist

3a. Set up the computer lab and method for students to submit their assignments		
Get unity licenses	→ You can either (a) apply for Unity Educational license through the license grant program or (b) have students create individual Unity IDs.	□
Install Unity software in computer lab	→ Download Unity Hub and install Unity version 2020.3 LTS (including Visual Studio) on all of the computers in the lab, then test to make sure that (a) Unity opens successfully and (b) Visual Studio opens successfully.	□
Set up system for students to submit their work	→ Using your school's LMS, Google Classroom, or other system, make sure your virtual classroom is set up so that students can submit their work. Students can submit screenshots/screencasts of their projects (recommended) or submit .zip files of their Unity assets. → It is possible to use version control software like Github to track and evaluate students' projects. ○ Unity has a built-in version control tool called Unity Collaborate, but this will not work with Unity Edu licenses.	□

3b. Prepare to teach and connect with a support community

Schedule time for training	■ Regardless of the Pathway configuration you have chosen, it is recommended that you complete *at least* the first Mission of the online Pathway independently prior to the Pathway start date. This will take approximately 25 hours. ■ If you intend to do any teacher-led activities, it is also strongly recommended that you complete that content in the online Pathway independently prior to leading the students.	□
Connect with the Unity teacher community	■ Click on this link to register and join the teacher support community, where you can get help from experts and connect with other new teachers.	□

3b. If relevant, purchase licenses for the Unity Certified User Exam		
Purchase exams from Certiport	■ If you intend on having students attempt the Unity Certified User Exam after the course, you need to purchase licenses for this exam from Certiport. ■ Note: this is only recommended if you are able to complete the entire Pathway.	□