## 



Unity Jr. Programmer Pathway

**Standards Alignment**

**International Society for Technology in Education (ISTE)**

*From the* [*ISTE Standards webpage*](https://www.iste.org/standards)*: The ISTE Standards are a framework for students, educators, administrators, coaches, and computer science educators to rethink education and create innovative learning environments.*

*Yes-* ✓*, No-* x*, Partial-* ◐

| **Domain** | **#** | **Standard** | |  |
| --- | --- | --- | --- | --- |
| **1**  **Empowered learner** | 1a | Students articulate and set personal learning goals, develop strategies leveraging technology to achieve them, and reflect on the learning process itself to improve learning outcomes. | | ◐ |
| 1c | Students use technology to seek feedback that informs and improves their practice and to demonstrate their learning in a variety of ways. | | ✓ |
| 1d | Students understand the fundamental concepts of technology operations, demonstrate the ability to choose, use, and troubleshoot current technologies, and are able to transfer their knowledge to explore emerging technologies. | | ✓ |
| **3**  **Knowledge constructor** | 3b | Students evaluate the accuracy, perspective, credibility, and relevance of information, media, data, or other resources. | | ✓ |
| 3c | Students curate information from digital resources using a variety of tools and methods to create collections of artifacts that demonstrate meaningful connections or conclusions. | | ◐ |
| 3d | Students build knowledge by actively exploring real-world issues and problems, developing ideas and theories, and pursuing answers and solutions. | | ✓ |
| **4**  **Innovative designer** | 4a | Students know and use a deliberate design process for generating ideas, testing theories, creating innovative artifacts, or solving authentic problems. | | ✓ |
| 4b | Students select and use digital tools to plan and manage a design process that considers design constraints and calculated risks. | | ✓ |
| 4c | Students develop, test, and refine prototypes as part of a cyclical design process. | | ✓ |
| 4d | Students exhibit a tolerance for ambiguity, perseverance, and the capacity to work with open-ended problems. | | ✓ |
| **5**  **Computational thinker** | 5a | Students formulate problem definitions suited for technology-assisted methods such as data analysis, abstract models, and algorithmic thinking in exploring and finding solutions. | | ◐ |
| 5c | Students break problems into component parts, extract key information, and develop descriptive models to understand complex systems or facilitate problem-solving. | | ◐ |
| **6**  **Creative communicator** | 6b | Students create original works or responsibly repurpose or remix digital resources into new creations. | | ✓ |
| **7**  **Global collaborator** | 7b | Students use collaborative technologies to work with others, including peers, experts, or community members, to examine issues and problems from multiple viewpoints. | | ◐ |



## [Unity Certified User: Programmer](https://unity.com/products/unity-certifications/user-programmer)

Future creators, start here on your path to a career within the real-time 3D ecosystem. Test your foundational Unity and C# programming skills, and tell the world that you’re ready to create games and apps in Unity.

*Yes-* ✓*, No-* x*, Partial-* ◐

| **Domain** |  |  |  |
| --- | --- | --- | --- |
| **Debugging, problem-solving, and interpreting the API** | Given an example of a debug log message, create the code that created the log message. | [Unit 1 - Player control](https://learn.unity.com/project/unit-1-driving-simulation?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f71fe63edbc2a00200e9de0) | ✓ |
| Given a code clip and its associated error message(s), determine which  object(s) is(are) null. | [Profile code to identify issues](https://learn.unity.com/tutorial/profile-code-to-identify-issues?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f779f1eedbc2a00201f3e5e) | ✓ |
| Given a specific programming task requiring the use of a particular class in the API, determine the appropriate method and/or properties, arguments, or  other syntax to use. | [Unit 4 - Gameplay Mechanics](https://learn.unity.com/project/unit-4-gameplay-mechanics?pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f7648a4edbc2a5578eb67df)  [Unit 3 - Sound and effects](https://learn.unity.com/project/unit-3-sound-and-effects?pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f7648a4edbc2a5578eb67df) | ✓ |
| **Creating code** | Indicate when and how to initialize and use variables including but not limited to the appropriate use of all variable modifiers and data collections such as Arrays, Lists, and Dictionaries. | [Unit 1 - Player control](https://learn.unity.com/project/unit-1-driving-simulation?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f71fe63edbc2a00200e9de0)  [Unit 3 - Sound and effects](https://learn.unity.com/project/unit-3-sound-and-effects?pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f7648a4edbc2a5578eb67df)  [Unit 4 - Gameplay Mechanics](https://learn.unity.com/project/unit-4-gameplay-mechanics?pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f7648a4edbc2a5578eb67df) | ◐ |
| Given a list of keywords and syntax elements, construct a viable Function declaration. | [Unit 1 - Player Control - Unity Learn](https://learn.unity.com/project/unit-1-driving-simulation?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f71fe63edbc2a00200e9de0)  [Unit 2 - Introduction - Unity Learn](https://learn.unity.com/tutorial/unit-2-introduction?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f71fe63edbc2a00200e9de0&projectId=5cdcc312edbc2a24a41671e6)  [Unit 4 - Gameplay Mechanics](https://learn.unity.com/project/unit-4-gameplay-mechanics?pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f7648a4edbc2a5578eb67df)  [Unit 3 - Sound and effects](https://learn.unity.com/project/unit-3-sound-and-effects?pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f7648a4edbc2a5578eb67df) | ✓ |
| Given a code clip and a description of its desired result, identify the appropriate function to control or trigger a state including but not limited to the Animator Controller. | [Unit 1 - Player Control - Unity Learn](https://learn.unity.com/project/unit-1-driving-simulation?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f71fe63edbc2a00200e9de0)  [Unit 2 - Introduction - Unity Learn](https://learn.unity.com/tutorial/unit-2-introduction?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f71fe63edbc2a00200e9de0&projectId=5cdcc312edbc2a24a41671e6)  [Unit 4 - Gameplay Mechanics](https://learn.unity.com/project/unit-4-gameplay-mechanics?pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f7648a4edbc2a5578eb67df)  [Unit 3 - Sound and effects](https://learn.unity.com/project/unit-3-sound-and-effects?pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f7648a4edbc2a5578eb67df) | ✓ |
| Given a scenario where a specific type of input is required and the building blocks needed are provided, construct the necessary input listener including but not limited to the keyboard and touch input. | [Unit 1 - Player Control](https://learn.unity.com/project/unit-1-driving-simulation?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f71fe63edbc2a00200e9de0)  [Unit 4 - Gameplay Mechanics](https://learn.unity.com/project/unit-4-gameplay-mechanics?pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f7648a4edbc2a5578eb67df)  [Unit 3 - Sound and effects](https://learn.unity.com/project/unit-3-sound-and-effects?pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f7648a4edbc2a5578eb67df) | ✓ |
| Demonstrate when and/or how to use the various logic and flow control operators used in C# and Unity. | [Unit 2 - Introduction](https://learn.unity.com/tutorial/unit-2-introduction?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f71fe63edbc2a00200e9de0&projectId=5cdcc312edbc2a24a41671e6)  [Unit 4 - Gameplay Mechanics](https://learn.unity.com/project/unit-4-gameplay-mechanics?pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f7648a4edbc2a5578eb67df)  [Unit 3 - Sound and effects](https://learn.unity.com/project/unit-3-sound-and-effects?pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f7648a4edbc2a5578eb67df) | ✓ |
| Given a scenario, identify appropriate actions to take when a UI element reports a change. | [Unit 4 - Gameplay Mechanics](https://learn.unity.com/project/unit-4-gameplay-mechanics?pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f7648a4edbc2a5578eb67df)  [Unit 3 - Sound and effects](https://learn.unity.com/project/unit-3-sound-and-effects?pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f7648a4edbc2a5578eb67df) | ✓ |
| **Evaluating code** | Given a scenario about the need to manage an event function, determine the appropriate action to take including but not limited to the keyboard and touch input | [Profile code to identify issues](https://learn.unity.com/tutorial/profile-code-to-identify-issues?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f779f1eedbc2a00201f3e5e) | ✓ |
| Given a code clip that produces an error because of a variable whose data type is declared incorrectly, identify the error | [Profile code to identify issues](https://learn.unity.com/tutorial/profile-code-to-identify-issues?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f779f1eedbc2a00201f3e5e) | ✓ |
| Given a code clip that produces an error because a function or variable is declared or used incorrectly (public/private mismatch), identify the error including but not limited to the use of Animation events | [Profile code to identify issues](https://learn.unity.com/tutorial/profile-code-to-identify-issues?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f779f1eedbc2a00201f3e5e) | ✓ |
| Given a code clip containing a class definition, distinguish whether the class is an ECS class or some other type of class. | [Profile code to identify issues](https://learn.unity.com/tutorial/profile-code-to-identify-issues?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f779f1eedbc2a00201f3e5e) | ◐ |
| Given a set of code clips, recognize the clip that uses naming conventions that observe Unity naming standards | [Profile code to identify issues](https://learn.unity.com/tutorial/profile-code-to-identify-issues?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f779f1eedbc2a00201f3e5e) | ✓ |
| Given a code clip (or a set of code clips), recognize the comments that  accurately describe what the code is doing. | [Profile code to identify issues](https://learn.unity.com/tutorial/profile-code-to-identify-issues?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f779f1eedbc2a00201f3e5e) | ✓ |
| **Navigating the Interface** | Describe the purpose, features, and functions of the various Unity IDE  windows. |  | ◐ |
| Demonstrate how to change the default scripting IDE. |  | x |
| Given a scenario that includes the following, then create a functional state machine.   1. a limited portion of a gaming scenario 2. a set of animation clips 3. a list of property settings | [Unit 2 - Introduction](https://learn.unity.com/tutorial/unit-2-introduction?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f71fe63edbc2a00200e9de0&projectId=5cdcc312edbc2a24a41671e6)  [Unit 3 - Sound and effects](https://learn.unity.com/project/unit-3-sound-and-effects?pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f7648a4edbc2a5578eb67df)  [Unit 4 - Gameplay mechanics](https://learn.unity.com/project/unit-4-gameplay-mechanics?pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f7648a4edbc2a5578eb67df) | ✓ |
| Create and program a function state machine within the Unity Animator Controller including but not limited to the use of Animator functions syntax |  | x |



[**Unity Certified Associate: Programmer**](https://unity.com/products/unity-certifications/associate-programmer?msclkid=47056e2eaeaf11ec99a72d5aef05b728)

Demonstrate core skills and competencies across programming, UI, debugging and asset management to help you obtain your first professional programming role with Unity. [Unity Certified Associate: Programmer](https://unity.com/products/unity-certifications/associate-programmer?msclkid=47056e2eaeaf11ec99a72d5aef05b728)

*Yes-* ✓*, No-* x*, Partial-* ◐

| **Domain** |  |  |  |
| --- | --- | --- | --- |
| **Unity Programming** | Evaluate code for integration into an existing system created/architected by a lead | * [Profile code to identify issues](https://learn.unity.com/tutorial/profile-code-to-identify-issues?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f779f1eedbc2a00201f3e5e) | ◐ |
| Make decisions required to prototype new concepts | * [Abstraction in object-oriented programming](https://learn.unity.com/tutorial/abstraction-in-object-oriented-programming?pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f779f1eedbc2a00201f3e5e) * [Inheritance and polymorphism in object-oriented programming](https://learn.unity.com/tutorial/inheritance-and-polymorphism-in-object-oriented-programming?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f779f1eedbc2a00201f3e5e) * [Encapsulation in object-oriented programming](https://learn.unity.com/tutorial/encapsulation-in-object-oriented-programming?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f779f1eedbc2a00201f3e5e) | ◐ |
| Determine code that would accomplish a specified interaction or programming logic | * [Unit 4 - Gameplay Mechanics](https://learn.unity.com/project/unit-4-gameplay-mechanics?pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f7648a4edbc2a5578eb67df) | ✓ |
| Decide how to implement scene management and transitions | * [Create a scene flow](https://learn.unity.com/tutorial/create-a-scene-flow?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f751af7edbc2a0022cdbbb6) * [Implement data persistence between scenes](https://learn.unity.com/tutorial/implement-data-persistence-between-scenes?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f751af7edbc2a0022cdbbb6) * [Implement data persistence between sessions](https://learn.unity.com/tutorial/implement-data-persistence-between-sessions?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f751af7edbc2a0022cdbbb6) | ✓ |
| Apply basic data persistence within a runtime session | * [Create a scene flow](https://learn.unity.com/tutorial/create-a-scene-flow?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f751af7edbc2a0022cdbbb6) * [Implement data persistence between scenes](https://learn.unity.com/tutorial/implement-data-persistence-between-scenes?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f751af7edbc2a0022cdbbb6) * [Implement data persistence between sessions](https://learn.unity.com/tutorial/implement-data-persistence-between-sessions?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f751af7edbc2a0022cdbbb6) | ✓ |
| Given a situation, determine proper usage and application of the Unity API | * [Abstraction in object-oriented programming](https://learn.unity.com/tutorial/abstraction-in-object-oriented-programming?pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f779f1eedbc2a00201f3e5e) * [Inheritance and polymorphism in object-oriented programming](https://learn.unity.com/tutorial/inheritance-and-polymorphism-in-object-oriented-programming?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f779f1eedbc2a00201f3e5e) * [Encapsulation in object-oriented programming](https://learn.unity.com/tutorial/encapsulation-in-object-oriented-programming?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f779f1eedbc2a00201f3e5e) | ✓ |
| Decide the appropriate properties, scripts, and components of GameObjects for required tasks | * [Unit 4 - Gameplay Mechanics](https://learn.unity.com/project/unit-4-gameplay-mechanics?pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f7648a4edbc2a5578eb67df) | ✓ |
| Choose the appropriate data structures for a specific situation | * [Abstraction in object-oriented programming](https://learn.unity.com/tutorial/abstraction-in-object-oriented-programming?pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f779f1eedbc2a00201f3e5e) * [Inheritance and polymorphism in object-oriented programming](https://learn.unity.com/tutorial/inheritance-and-polymorphism-in-object-oriented-programming?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f779f1eedbc2a00201f3e5e) * [Encapsulation in object-oriented programming](https://learn.unity.com/tutorial/encapsulation-in-object-oriented-programming?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f779f1eedbc2a00201f3e5e) | ◐ |
| Choose the appropriate data types for a specific situation | * [Unit 2 - Introduction - Unity Learn](https://learn.unity.com/tutorial/unit-2-introduction?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f71fe63edbc2a00200e9de0&projectId=5cdcc312edbc2a24a41671e6) * [Unit 3 - Sound and effects](https://learn.unity.com/project/unit-3-sound-and-effects?pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f7648a4edbc2a5578eb67df) * [Unit 4 - Gameplay Mechanics](https://learn.unity.com/project/unit-4-gameplay-mechanics?pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f7648a4edbc2a5578eb67df) | ◐ |
| Identify the steps required to deploy a basic build | * [Unit 2 - Introduction - Unity Learn](https://learn.unity.com/tutorial/unit-2-introduction?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f71fe63edbc2a00200e9de0&projectId=5cdcc312edbc2a24a41671e6) * [Unit 3 - Sound and effects](https://learn.unity.com/project/unit-3-sound-and-effects?pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f7648a4edbc2a5578eb67df) * [Unit 4 - Gameplay Mechanics](https://learn.unity.com/project/unit-4-gameplay-mechanics?pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f7648a4edbc2a5578eb67df) | ◐ |
| Apply concepts required to write code with basic inheritance and interfaces | * [Abstraction in object-oriented programming](https://learn.unity.com/tutorial/abstraction-in-object-oriented-programming?pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f779f1eedbc2a00201f3e5e) * [Inheritance and polymorphism in object-oriented programming](https://learn.unity.com/tutorial/inheritance-and-polymorphism-in-object-oriented-programming?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f779f1eedbc2a00201f3e5e) * [Encapsulation in object-oriented programming](https://learn.unity.com/tutorial/encapsulation-in-object-oriented-programming?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f779f1eedbc2a00201f3e5e) | ◐ |
| **UI** | Apply concepts required to lay out a user interface | * [Unit 5 - User Interface](https://learn.unity.com/project/unit-5-user-interface?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f7648a4edbc2a5578eb67df) | ✓ |
| Identify the process required to bind data on the UI to application data | * [Unit 5 - User Interfac](https://learn.unity.com/project/unit-5-user-interface?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f7648a4edbc2a5578eb67df) | ✓ |
| Decide how to capture and respond to UI input using the Event System | * [Unit 2 - Introduction - Unity Learn](https://learn.unity.com/tutorial/unit-2-introduction?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f71fe63edbc2a00200e9de0&projectId=5cdcc312edbc2a24a41671e6) * [Unit 3 - Sound and effects](https://learn.unity.com/project/unit-3-sound-and-effects?pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f7648a4edbc2a5578eb67df) * [Unit 4 - Gameplay Mechanics](https://learn.unity.com/project/unit-4-gameplay-mechanics?pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f7648a4edbc2a5578eb67df) | ✓ |
| Decide how to capture and respond to UI input using the Event System | * [Unit 4 - Gameplay Mechanics](https://learn.unity.com/project/unit-4-gameplay-mechanics?pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f7648a4edbc2a5578eb67df) | ✓ |
| **Debugging** | Troubleshoot code that fails to perform as expected | * [Profile code to identify issues](https://learn.unity.com/tutorial/profile-code-to-identify-issues?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f779f1eedbc2a00201f3e5e) | ✓ |
| Troubleshoot common compilation bugs | * [Profile code to identify issues](https://learn.unity.com/tutorial/profile-code-to-identify-issues?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f779f1eedbc2a00201f3e5e) | ✓ |
| Troubleshoot runtime exceptions | * [Profile code to identify issues](https://learn.unity.com/tutorial/profile-code-to-identify-issues?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f779f1eedbc2a00201f3e5e) | ◐ |
| Determine techniques required to refactor and improve code | * [Profile code to identify issues](https://learn.unity.com/tutorial/profile-code-to-identify-issues?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f779f1eedbc2a00201f3e5e) | ◐ |
| Determine techniques required to profile and debug trivial performance issues | * [Profile code to identify issues](https://learn.unity.com/tutorial/profile-code-to-identify-issues?uv=2020.3&pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f779f1eedbc2a00201f3e5e) | ◐ |
| **Asset Management** | Identify the process required to create a prefab from art and code | * [Unit 4 - Gameplay Mechanics](https://learn.unity.com/project/unit-4-gameplay-mechanics?pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f7648a4edbc2a5578eb67df) | ✓ |
| Identify properties of nested prefabs and prefab variants | * [Unit 4 - Gameplay Mechanics](https://learn.unity.com/project/unit-4-gameplay-mechanics?pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f7648a4edbc2a5578eb67df) | ✓ |
| Identify the primary purposes of version control when working with Unity | * [Set up version control](https://learn.unity.com/tutorial/set-up-version-control?pathwayId=5f7e17e1edbc2a5ec21a20af&missionId=5f751af7edbc2a0022cdbbb6) | ✓ |