# Coding in Minecraft - Python Expert Coding – Exam Objective Domain

| ID | Exam & Learning Objectives | Evidence | CSTA Standards Alignment | AP CSP Learning Objectives Alignment |
|---|---|---|---|---|
| **1** | **Variables & Data Types** | | | |
| 1.1 | Identify and use variables in Python and assign appropriate data types | Be able to define a variable using Python<br>Be able to choose an appropriate data type of str, int, float or bool and explain why | 3B-AP-12 Compare and contrast fundamental data structures and their uses. | AAP-1.A Represent a value with a variable. 3.A |
| 1.2 | Convert between data types | Be able to convert from one data type to another | 3B-AP-12 Compare and contrast fundamental data structures and their uses. | AAP-1.B Determine the value of a variable because of an assignment. 4.B |
| 1.3 | Identify and use Operators | Be able to identify and use Operators in Python - assignment, comparison, logical, arithmetic, is and in<br>Be able to determine operator precedence and sequence of execution | 3B-AP-12 Compare and contrast fundamental data structures and their uses. | AAP-2.C Evaluate expressions that use arithmetic operators. 4.B<br><br>AAP-2.E For relationships between two variables, expressions, or values:<br>a. Write expressions using relational operators. 2.B<br>b. Evaluate expressions that use relational operators. 4.B |


© Prodigy Learning | prodigylearning.com | codingcredentials.com

| ID | Exam & Learning Objectives | Evidence | CSTA Standards Alignment | AP CSP Learning Objectives Alignment |
|---|---|---|---|---|
| **2** | **Selection/Branching - If Statements** | | | |
| 2.1 | Identify and build If statements | Be able to identify and build code using if statements | 3A-AP-15 Justify the selection of specific control structures when tradeoffs involve implementation, readability, and program performance, and explain the benefits and drawbacks of choices made. | AAP-2.H For selection:<br>a. Write conditional statements. 2.B<br>b. Determine the result of conditional statements. 4.B |
| 2.2 | Identify and build complex if statements | Be able to identify and build code using elif and else statements to create complex selections | 3A-AP-15 Justify the selection of specific control structures when tradeoffs involve implementation, readability, and program performance, and explain the benefits and drawbacks of choices made. | AAP-2.I For nested selection:<br>a. Write nested conditional statements 2.B<br>b. Determine the result of nested conditional statements. 4.B |
| 2.3 | Identify and use If statements that contain compound conditionals | Be able to identify and build code using complex compound conditionals (Or, And) and Not operator | 3A-AP-15 Justify the selection of specific control structures when tradeoffs involve implementation, readability, and program performance, and explain the benefits and drawbacks of choices made. | AAP-2.F For relationships betweenBoolean values: a. Write expressions using logical operators. 2.Bb. Evaluate expressions that use logic operators. 4.B |


© Prodigy Learning | prodigylearning.com | codingcredentials.com

| ID | Exam & Learning Objectives | Evidence | CSTA Standards Alignment | AP CSP Learning Objectives Alignment |
|---|---|---|---|---|
| **3** | **Iteration - Loops** | | | |
| 3.1 | Identify and build while loops | Be able to identify and build code which uses while loop statements | 3A-AP-15 Justify the selection of specific control structures when tradeoffs involve implementation, readability, and program performance, and explain the benefits and drawbacks of choices made. | AAP-2.K For iteration:<br>a. Write iteration statements. 2.B<br>b. Determine the result or side effect of iteration statements. 4.B |
| 3.2 | Identify and build for loops | Be able to identify and build code which uses for loop statements | 3A-AP-15 Justify the selection of specific control structures when tradeoffs involve implementation, readability, and program performance, and explain the benefits and drawbacks of choices made. | AAP-2.K For iteration:<br>a. Write iteration statements. 2.B<br>b. Determine the result or side effect of iteration statements. 4.B |
| 3.3 | Identify and build nested loops | Be able to identify and build code which uses nested loops | 3A-AP-15 Justify the selection of specific control structures when tradeoffs involve implementation, readability, and program performance, and explain the benefits and drawbacks of choices made. | AAP-2.K For iteration:<br>a. Write iteration statements. 2.B<br>b. Determine the result or side effect of iteration statements. 4.B |
| 3.4 | Identify and build loops that use complex compound conditionals | Be able to identify and build loops that use complex compound conditionals (Or, And, Not) | 3A-AP-15 Justify the selection of specific control structures when tradeoffs involve implementation, readability, and program performance, and explain the benefits and drawbacks of choices made. | AAP-2.F For relationships between Boolean values:<br>a. Write expressions using logical operators. 2.B<br>b. Evaluate expressions that use logic operators. 4.B |


© Prodigy Learning | prodigylearning.com | codingcredentials.com

| ID | Exam & Learning Objectives | Evidence | CSTA Standards Alignment | AP CSP Learning Objectives Alignment |
|---|---|---|---|---|
| 3.5 | Identify and use loop controls | Be able to identify, understand and implement continue, break, pass statements in loops | 3A-AP-15 Justify the selection of specific control structures when tradeoffs involve implementation, readability, and program performance, and explain the benefits and drawbacks of choices made. | AAP-2.K For iteration:<br>a. Write iteration statements. 2.B<br>b. Determine the result or side effect of iteration statements. 4.B |


© Prodigy Learning | prodigylearning.com | codingcredentials.com

| ID | Exam & Learning Objectives | Evidence | CSTA Standards Alignment | AP CSP Learning Objectives Alignment |
|---|---|---|---|---|
| **4** | **Collections** | | | |
| 4.1 | Identify and use lists | Be able to identify, understand when to use and build code which uses lists (creation, indexing, modifying, traversing, common methods) | 3A-AP-14 Use lists to simplify solutions, generalizing computational problems instead of repeatedly using simple variables. | AAP-1.C Represent a list or string using a variable. 3.A<br><br>AAP-1.D For data abstraction:<br>a. Develop data abstraction using lists to store multiple elements 3.B<br>b. Explain how the use of data abstraction manages complexity in program code. 3.C<br><br>AAP-2.N For list operations:<br>a. Write expressions that use list indexing and list procedures. 2.B<br>b. Evaluate expressions that use list indexing and list procedures. 4.B |
| 4.2 | Identify and use tuples | Be able to identify, understand when to use and build code which uses tuples (creation, unpacking, iterating, common methods) | 3A-AP-14 Use lists to simplify solutions, generalizing computational problems instead of repeatedly using simple variables. | AAP-1.C Represent a list or string using a variable. 3.A<br><br>AAP-1.D For data abstraction:<br>a. Develop data abstraction using lists to store multiple elements 3.B<br>b. Explain how the use of data abstraction manages complexity in program code. 3.C<br><br>AAP-2.N For list operations:<br>a. Write expressions that use list indexing and list procedures. 2.B<br>b. Evaluate expressions that use list indexing and list procedures. 4.B |


© Prodigy Learning | prodigylearning.com | codingcredentials.com

| ID | Exam & Learning Objectives | Evidence | CSTA Standards Alignment | AP CSP Learning Objectives Alignment |
|---|---|---|---|---|
| 4.3 | Identify and use dictionaries | Be able to identify, understand when to use and build code which uses dictionaries (creation, modifying, membership, iteration, common methods) | 3A-AP-14 Use lists to simplify solutions, generalizing computational problems instead of repeatedly using simple variables. | AAP-1.C Represent a list or string using a variable. 3.A<br><br>AAP-1.D For data abstraction:<br>a. Develop data abstraction using lists to store multiple elements 3.B<br>b. Explain how the use of data abstraction manages complexity in program code. 3.C<br><br>AAP-2.N For list operations:<br>a. Write expressions that use list indexing and list procedures. 2.B<br>b. Evaluate expressions that use list indexing and list procedures. 4.B |


© Prodigy Learning | prodigylearning.com | codingcredentials.com

| ID | Exam & Learning Objectives | Evidence | CSTA Standards Alignment | AP CSP Learning Objectives Alignment |
|---|---|---|---|---|
| **5** | **Structure and document code appropriately** | | | |
| 5.1 | Use whitespace and indentation | Be able to correctly use whitespace and indentation to structure code appropriately | 3A-AP-21 Evaluate and refine computational artifacts to make them more usable and accessible.<br><br>3A-AP-23 Document design decisions using text, graphics, presentations, and/or demonstrations in the development of complex programs. | CRD-2.G Describe the purpose of a code segment or program by writing documentation. 4.A |
| 5.2 | Document code | Be able to document code using comments and documentation stings and generate documentation using pydoc | 3A-AP-21 Evaluate and refine computational artifacts to make them more usable and accessible.<br><br>3B-AP-20 Use version control systems, integrated development environments (IDEs), and collaborative tools and practices (code documentation) in a group software project. | CRD-2.G Describe the purpose of a code segment or program by writing documentation. 4.A<br><br>CRD-2.H Acknowledge code segments used from other sources. 1.C |
| 5.3 | Code review | Be able to perform a code review of another person's code | 3B-AP-23 Evaluate key qualities of a program through a process such as a code review. | |


© Prodigy Learning | prodigylearning.com | codingcredentials.com

| ID | Exam & Learning Objectives | Evidence | CSTA Standards Alignment | AP CSP Learning Objectives Alignment |
|---|---|---|---|---|
| **6** | **Error handling** | | | |
| 6.1 | Understand the different types of errors and causes | Be able to identify, understand, differentiate, and perform basic troubleshooting procedures and fix syntax errors, semantic/logic error, and runtime errors | 3B-AP-18 Explain security issues that might lead to compromised computer programs. | CRD-2.I For errors in an algorithm or program:<br>a. Identify the error. 4.C<br>b. Correct the error. 4.C |
| 6.2 | Understand code testing and create tests | Be able to understand the need for and importance of testing and be able to create tests to validate code | 3B-AP-21 Develop and use a series of test cases to verify that a program performs according to its design specifications. | CRD-2.J Identify inputs and corresponding expected outputs or behaviours that can be used to check the correctness of an algorithm<br>or program. 4.C |
| 6.3 | Identify and build code that handles errors and exceptions | Be able to identify, understand when to use and build code which uses exception handling - try, else, finally, raise, except | 3B-AP-18 Explain security issues that might lead to compromised computer programs. | |


© Prodigy Learning | prodigylearning.com | codingcredentials.com

| ID | Exam & Learning Objectives | Evidence | CSTA Standards Alignment | AP CSP Learning Objectives Alignment |
|---|---|---|---|---|
| **7** | **Modularity** | | | |
| 7.1 | Understand modularity and identify when to use | Be able to understand the concept of modularity and identify when modularity can or should be used in code | 3A-AP-17 Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.<br><br>3B-AP-15 Analyse a large-scale computational problem and identify generalizable patterns that can be applied to a solution. | AAP-3.B Explain how the use of procedural abstraction manages complexity in a program. 3.C<br><br>CRD-2.B Explain how a program or code segment functions. 4.A |
| 7.2 | Identify and build functions | Be able to identify, understand, use (call) and build functions - def statement | 3A-AP-16 Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue by using events to initiate instructions.<br><br>3A-AP-18 Create artifacts by using procedures within a program, combinations of data and procedures, or independent but interrelated programs.<br><br>3B-AP-14 Construct solutions to problems using student-created components, such as procedures, modules and/or objects. | AAP-3.A For procedure calls:<br>a. Write statements to call procedures. 3.B<br>b. Determine the result or effect of a procedure call. 4.B<br><br>AAP-3.C Develop procedural abstractions to manage complexity in a program by writing procedures. 3.B |


© Prodigy Learning | prodigylearning.com | codingcredentials.com

| ID | Exam & Learning Objectives | Evidence | CSTA Standards Alignment | AP CSP Learning Objectives Alignment |
|---|---|---|---|---|
| 7.3 | Identify and build functions that take or return data | Be able to identify, understand, use (call) and build functions that take parameters or return data - passing parameters/arguments (inc optional and keyword arguments) and return statement | 3A-AP-16 Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue by using events to initiate instructions.<br><br>3A-AP-18 Create artifacts by using procedures within a program, combinations of data and procedures, or independent but interrelated programs.<br><br>3B-AP-14 Construct solutions to problems using student-created components, such as procedures, modules and/or objects. | AAP-3.A For procedure calls:<br>a. Write statements to call procedures. 3.B<br>b. Determine the result or effect of a procedure call. 4.B<br><br>AAP-3.C Develop procedural abstractions to manage complexity in a program by writing procedures. 3.B |


© Prodigy Learning | prodigylearning.com | codingcredentials.com

| ID | Exam & Learning Objectives | Evidence | CSTA Standards Alignment | AP CSP Learning Objectives Alignment |
|---|---|---|---|---|
| **8** | **Input and Output** | | | |
| 8.1 | Identify and build code that performs console input operations | Be able to identify and build code which reads input from the console and stores or processes appropriately | | |
| 8.2 | Identify and build code that performs console output operations | Be able to identify and build code that outputs to the console and format output appropriately | | |
| 8.3 | Identify and build code that performs file input operations | Be able to identify and build code that opens, reads, and closes text files including using the with statement | | |


© Prodigy Learning | prodigylearning.com | codingcredentials.com