**SYNTAX TEST SCRIPT**

**General Structure of the Program**Syntax: <program> → entrance <global\_dec><global\_funcdec> <mane> <sub\_function> exit

|  |  |  |  |
| --- | --- | --- | --- |
| Sample Input | Expected Output | Actual Output | Remarks |
| entrance  mane()  {{  zoout((“Hello”:  }}  exit | No Error |  |  |
| entrance  mane()  {{  }}  exit | Unexpected token “}}”, expected one of “zooin”, “zoout”, “id”, “if” “swasp”, “whale”, “do”, or “fur” |  |  |
| mane()  {{  zoout((“Hello”:  }}  exit | Unexpected token “mane”, expected “entrance” |  |  |
| entrance  mane(  {{  zoout((“Hello”:  }}  exit | Unexpected token “{{“, expected “)” |  |  |
| entrance  mane()  zoout((“Hello”:  }}  exit | Unexpected token “zoout”, expected “{{“ |  |  |
| entrance  mane()  {{  zoout((“Hello”  }}  exit | Unexpected token “}}”, expected “:” |  |  |
| entrance  mane()  {{  zoout((  }}  exit |  |  |  |

**Constant Declaration**Syntax: <const\_dec> → let <ident\_var> = <val><const\_next>:<comment>

|  |  |  |  |
| --- | --- | --- | --- |
| Sample Input | Expected Output | Actual Output | Remarks |
| let newt at $age = 19: | No Error |  |  |
| let newt at $age = 15, newt at $num = 20: | No Error |  |  |
| let newt at $age, $num = 45: | unexpected token ",", expected "=", on line |  |  |
| Let Int Age 25. |  |  |  |
| Let Double Grade  is 1.5. |  |  |  |
| Let Double Grade is 1.5; Mile is 2.5. |  |  |  |
| Let Double Grade; Mile is 2.5. |  |  |  |
| Let Double Grade 1.5. |  |  |  |
| Let Char Letter is ‘A’. |  |  |  |
| Let Char Letter is ‘A’; Vowel is ‘E’. |  |  |  |
| Let Char Letter; Vowel is ‘E’. |  |  |  |
| Let Char Letter ‘A’. |  |  |  |
| Let String Name is “Joe”. |  |  |  |
| Let String Name is “Joe”; Surname is “Sam”. |  |  |  |
| Let String Name; Surname is “Sam”. |  |  |  |
| Let String Name “Joe”. |  |  |  |
| Let Boolean Case is No. |  |  |  |
| Let Boolean Case is No; Status is Yes. |  |  |  |
| Let Boolean Case; Status is Yes. |  |  |  |
| Let Boolean Case No. |  |  |  |

**Variable Declaration**Syntax: <vardec> → <ident\_var><next2var>:

|  |  |  |  |
| --- | --- | --- | --- |
| Sample Input | Expected Output | Actual Output | Remarks |
| Var Int Age is 25. | No Error | **No Error** | **OK** |
| Var Int Age is 35; Num is 45. |  |  |  |
| Var Int Age; Num is 55. |  |  |  |
| Var Int Age 25. |  |  |  |
| Var Double Grade  is 1.5. |  |  |  |
| Var Double Grade is 1.5; Mile is 2.5. |  |  |  |
| Var Double Grade; Mile is 2.5. |  |  |  |
| Var Double Grade 1.5. |  |  |  |
| Var Char Letter is ‘A’. |  |  |  |
| Var Char Letter is ‘A’; Vowel is ‘E’. |  |  |  |
| Var Char Letter; Vowel is ‘E’. |  |  |  |
| Var Char Letter ‘A’. |  |  |  |
| Var String Name is “Joe”. |  |  |  |
| Var String Name is “Joe”; Surname is “Sam”. |  |  |  |
| Var String Name; Surname is “Sam”. |  |  |  |
| Var String Name “Joe”. |  |  |  |
| Var Boolean Case is No. |  |  |  |
| Var Boolean Case is No; Status is Yes. |  |  |  |
| Var Boolean Case; Status is Yes. |  |  |  |
| Var Boolean Case No. |  |  |  |

**Read Statement**Syntax: <input> → Read id

|  |  |  |  |
| --- | --- | --- | --- |
| Sample Input | Expected Output | Actual Output | Remarks |
| Read Age. | No Error | **No Error** | **OK** |
| Read Age; Num. |  |  |  |
| Read Age |  |  |  |
| Lead:  Start  Read Age.  End.# |  |  |  |
| Lead:  Read Age.  End. |  |  |  |

**Say Statement**Syntax: <input> → Read id

|  |  |  |  |
| --- | --- | --- | --- |
| Sample Input | Expected Output | Actual Output | Remarks |
| Read Age. | No Error | **No Error** | **OK** |
| Read Age; Num. |  |  |  |
| Read Age |  |  |  |
| Lead:  Start  Read Age.  End.# |  |  |  |
| Lead:  Read Age.  End.# |  |  |  |