Responses from the MCP Server

Tools

generate_mcq

Example 1 Input: `Generate multiple-choice questions on python basics`

Output:

```
{
  "questions": [
    {
      "question": "What is the correct way to print 'Hello, world!' in Python?",
      "answer": "print('Hello, world!')",
      "explanation": "The `print()` function is used to display output in Python.",
      "options": [
        "print('Hello, world!')",
        "echo 'Hello, world!'",
        "System.out.println('Hello, world!')",
        "printf('Hello, world!')"
      ]
    },
    {
      "question": "What data type is 10?",
      "answer": "integer",
      "explanation": "Numbers without decimal points are integers in Python.",
      "options": [
        "integer",
        "string",
        "float",
        "boolean"
      ]
    },
    {
      "question": "What data type is '10'?",
      "answer": "string",
      "explanation": "Values enclosed in quotes are strings.",
      "options": [
        "integer",
        "string",
        "float",
        "boolean"
      ]
    },
    {
```

```
    "question": "What data type is 10.5?",
    "answer": "float",
    "explanation": "Numbers with decimal points are floating-point numbers (floats).",
    "options": [
      "integer",
      "string",
      "float",
      "boolean"
    ]
  },
  {
    "question": "What is the result of 5 + 2?",
    "answer": "7",
    "explanation": "Basic arithmetic operation.",
    "options": [
      "7",
      "52",
      "10",
      "3"
    ]
  },
  {
    "question": "What is the result of 10 / 2?",
    "answer": "5.0",
    "explanation": "Division in Python always results in a float.",
    "options": [
      "5",
      "5.0",
      "2",
      "10/2"
    ]
  },
  {
    "question": "What does '#' denote in a Python code?",
    "answer": "Comment",
    "explanation": "A single-line comment starts with '#'.",
    "options": [
      "Variable",
      "Function",
      "Comment",
      "Operator"
    ]
  },
  {
    "question": "How do you assign the value 10 to a variable named 'x'?",
    "answer": "x = 10",
      "explanation": "Use the assignment operator '='.",
      "options": [
        "x = 10",
        "10 = x",
        "x == 10",
        "x := 10"
      ]
    },
    {
      "question": "What is the output of `type(True)`?",
      "answer": "<class 'bool'>",
      "explanation": "True and False are boolean values.",
      "options": [
        "<class 'int'>",
        "<class 'str'>",
        "<class 'bool'>",
        "<class 'float'>"
      ]
    },
    {
      "question": "What will be the output of `len('Python')`?",
      "answer": "6",
      "explanation": "`len()` function returns the length of a string.",
      "options": [
        "5",
        "6",
        "Python",
        "Error"
      ]
    }
  ]
}
```

Example 2 Input: `generate 5 mcqs on python basics in hard difficulty`

Output:

```
{
  "questions": [
    {
      "question": "Explain the difference between shallow copy and deep copy in Python, prov
      "answer": "A shallow copy creates a new object, but it populates it with references to
      "explanation": "The correct answer highlights the key difference and the implications
      "options": [
        "Shallow copy copies only the references, deep copy copies the values.",
        "Shallow copy is faster and always preferred.",
        "Deep copy is slower and only necessary for immutable objects.",
        "Shallow copy and deep copy are functionally identical."
      ]
    },
    {
      "question": "Describe a scenario where using a generator function in Python would sign
      "answer": "Generator functions are more memory-efficient when dealing with large datas
      "explanation": "The correct option emphasizes memory efficiency, the primary advantage
      "options": [
        "Generator functions are always faster than regular functions.",
        "Generator functions are only useful for small datasets.",
        "Generator functions and regular functions have identical performance for large data
        "Generator functions are more memory-efficient for large datasets."
      ]
    },
    {
      "question": "Explain the concept of decorators in Python and provide a practical examp
      "answer": "Decorators are a powerful and expressive feature in Python that allows you
      "explanation": "The correct answer accurately describes decorators and their use.  Inc
      "options": [
        "Decorators are only useful for simple functions.",
        "Decorators cannot handle functions with variable arguments.",
        "Decorators modify the original function's code directly.",
        "Decorators provide a way to add functionality to functions without modifying them."
      ]
    },
    {
      "question": "How can you effectively handle exceptions in Python to ensure robustness
      "answer": "Robust exception handling involves using `try-except-finally` blocks to cat
      "explanation": "The correct answer stresses the importance of specific exception handl
      "options": [
        "Exceptions should always be ignored to prevent program crashes.",
        "`try-except` blocks are sufficient for all exception handling scenarios.",
        "`finally` blocks are optional and rarely needed.",
        "Effective exception handling requires specific exception types and resource cleanup
      ]
    },
    {
      "question": "Discuss the differences between class methods, static methods, and instan
      "answer": "Instance methods operate on an instance of a class and have access to the i
      "explanation": "The correct answer clearly differentiates the three method types and t
      "options": [
        "All three method types are functionally equivalent.",
        "Class methods are always faster than instance methods.",
        "Static methods are only used for mathematical calculations.",
        "The choice of method type depends on access to instance and class attributes."
      ]
    }
  ]
}
```

generate_flashcard

Example 1 **Input**: `Generate flashcards about AI Basics`

Output: Published output at: `https://claude.ai/public/artifacts/945e3f7e-1754-4c16-ae38-7cb0d00`
or

```
{
  "title": "AI Basics",
  "flashcards": [
    {
      "front": "What is Artificial Intelligence (AI)?",
      "back": "Artificial intelligence is the simulation of human intelligence processes by
      "explanation": null
    },
    {
      "front": "Define Machine Learning (ML).",
      "back": "Machine learning is a subset of AI where systems learn from data without expl
      "explanation": null
    },
    {
      "front": "What is Deep Learning (DL)?",
      "back": "Deep learning is a subfield of machine learning that uses artificial neural n
      "explanation": null
    },
    {
      "front": "Explain the difference between supervised and unsupervised learning.",
      "back": "Supervised learning uses labeled data (input and desired output) to train a m
      "explanation": null
    },
    {
      "front": "What is a neural network?",
      "back": "A neural network is a computing system inspired by the biological neural netw
      "explanation": null
    },
    {
      "front": "What is reinforcement learning?",
      "back": "Reinforcement learning is a type of machine learning where an agent learns to
      "explanation": null
    },
    {
      "front": "What is a bias in AI?",
      "back": "Bias in AI refers to systematic and repeatable errors in a system's output ca
      "explanation": null
    },
    {
      "front": "What is natural language processing (NLP)?",
      "back": "Natural language processing (NLP) is a branch of AI that focuses on enabling
      "explanation": null
    },
    {
      "front": "What is computer vision?",
      "back": "Computer vision is a field of AI that enables computers to 'see' and interpre
      "explanation": null
    },
    {
      "front": "What are some ethical considerations in AI?",
      "back": "Ethical considerations in AI include bias, fairness, accountability, transpar
      "explanation": null
    }
  ]
}
```

Example 2 **Input**: `create flashcards on python programming`

Output:

```
{'title': 'Python Programming Fundamentals', 'flashcards': [{'front': 'What is Python?', 'ba
```

Resources

devise_lessonplan

Example **Input**: { “topic”: “algebra” }

Output: Can be seen only through debugging due to *lack of support for dynamic resource* from claude desktop(no support for resource template)

```
{
  "title": "Unlocking the Power of Algebra",
  "subject": "Algebra",
  "learning_objectives": [
    "Students will be able to define variables and constants and use them in simple algebrai
    "Students will be able to solve one-step and two-step linear equations.",
    "Students will be able to apply algebraic concepts to solve real-world problems and inte
  ],
  "lesson_introduction": "Imagine building a robot!  To make it move, you need precise instr
  "main_topics": [
    {
```

```
"title": "Understanding Variables and Expressions",
"subtopics": [
  {
    "title": "Variables and Constants",
    "key_concepts": [
      {
        "type": "definition",
        "content": "A variable is a symbol (usually a letter) that represents an unkno
      },
      {
        "type": "example",
        "content": "In the expression 2x + 5, 'x' is a variable and '2' and '5' are co
      },
      {
        "type": "illustration",
        "content": "Visual representation of variables and constants using different c
      }
    ],
    "discussion_questions": [
      {
        "question": "Why do we use variables in algebra?"
      },
      {
        "question": "Can a variable represent different values?"
      }
    ],
    "hands_on_activities": [
      {
        "title": "Variable Scavenger Hunt",
        "description": "Students identify real-world examples of variables (e.g., the
      }
    ],
    "reflective_questions": [
      {
        "question": "How can you distinguish between a variable and a constant?"
      }
    ],
    "assessment_ideas": [
      {
        "type": "quiz",
        "description": "Short quiz identifying variables and constants in given expres
      }
    ]
  },
  {
    "title": "Algebraic Expressions",
      "key_concepts": [
        {
          "type": "definition",
          "content": "An algebraic expression is a combination of variables, constants,
        },
        {
          "type": "example",
          "content": "Examples: 3a - 7,  x² + 2y,  (4 + b)/2"
        },
        {
          "type": "multimedia",
          "content": "Short video explaining how to build and simplify algebraic express
        }
      ],
      "discussion_questions": [
        {
          "question": "How can you simplify an algebraic expression?"
        },
        {
          "question": "What are the order of operations in algebra?"
        }
      ],
      "hands_on_activities": [
        {
          "title": "Expression Building Blocks",
          "description": "Students use blocks representing variables and constants to bu
        }
      ],
      "reflective_questions": [
        {
          "question": "What are the different components of an algebraic expression?"
        }
      ],
      "assessment_ideas": [
        {
          "type": "written task",
          "description": "Write and simplify several algebraic expressions."
        }
      ]
    }
  ]
},
{
  "title": "Solving Linear Equations",
  "subtopics": [
    {
      "title": "One-Step Equations",
      "key_concepts": [
        {
          "type": "definition",
          "content": "A one-step equation involves one operation (addition, subtraction,
        },
        {
          "type": "example",
          "content": "x + 5 = 10;  x - 3 = 7;  2x = 6;  x/4 = 2"
        },
        {
          "type": "illustration",
          "content": "Visual representation using a balance scale."
        }
      ],
      "discussion_questions": [
        {
          "question": "What is the inverse operation of addition? Subtraction? Multiplic
        },
        {
          "question": "How do you maintain balance when solving equations?"
        }
      ],
      "hands_on_activities": [
        {
          "title": "Equation Balance",
          "description": "Students use a balance scale to solve one-step equations."
        }
      ],
      "reflective_questions": [
        {
          "question": "Explain the steps involved in solving a one-step equation."
        }
      ],
      "assessment_ideas": [
        {
          "type": "quiz",
          "description": "Solve a set of one-step equations."
        }
      ]
    },
    {
      "title": "Two-Step Equations",
      "key_concepts": [
        {
          "type": "definition",
              "content": "A two-step equation involves two operations to solve for the varia
            },
            {
              "type": "example",
              "content": "2x + 3 = 7;  x/2 - 1 = 3"
            },
            {
              "type": "multimedia",
              "content": "Interactive online tool for solving two-step equations."
            }
          ],
          "discussion_questions": [
            {
              "question": "What is the order of operations when solving two-step equations?"
            },
            {
              "question": "How can you check your answer to an equation?"
            }
          ],
          "hands_on_activities": [
            {
              "title": "Equation Challenge",
              "description": "Students create and solve their own two-step equations."
            }
          ],
          "reflective_questions": [
            {
              "question": "How do you decide which operation to perform first when solving a
            }
          ],
          "assessment_ideas": [
            {
              "type": "project",
              "description": "Create a real-world problem that can be solved using a two-ste
            }
          ]
        }
      ]
    }
  ],
  "learning_adaptations": "For younger grades, focus on concrete examples and manipulatives.
  "real_world_applications": "Algebra is crucial in many fields, including engineering, comp
  "ethical_considerations": "The ethical use of algebra involves ensuring that mathematical
}
```

get_info (uri = ‘about://info’)

Example Input: NA

Output:

```
{
    "name": "Educhain - MCP server",
    "description": "A simple Educhain based MCP server that uses google gemini 1.5 flash to
    "version": "0.1.0",
    "tools": ["generate_mcq", "generate_flashcard"],
    "resources": ["list_resources", "get_info", "devise_lessonplan"],
    "creator": "Bandi Anudeep Reddy (github: Anudeep-CodeSpace)"
}
```

list_resources (uri = ‘list://resources’)

Input: NA

Output:

```
{
    'resources': [
        {
            "uri": "hello://world",
            "name": "Hello world resource",
            "description": "Sample hello world! resource for testing",
            "mime_type": "text/plain"
        },
        {
            "uri": "greetings://{name}",
            "name": "Send greetings",
            "description": "Send personalized greetings to a given name",
            "mime_type": "text/plain"
        },
        {
            "uri": "lessonplan://{subject}",
            "name": "Device lesson plan",
            "description": "Device a lesson plan for a given subject",
            "mime_type": "LessonPlanResponse",
            "examples": [
                "lessonplan://algebra",
                "lessonplan://math"
            ]
        }
    ]
}
```