



ACADEMY
OF DIGITAL ARTS
EGYPT



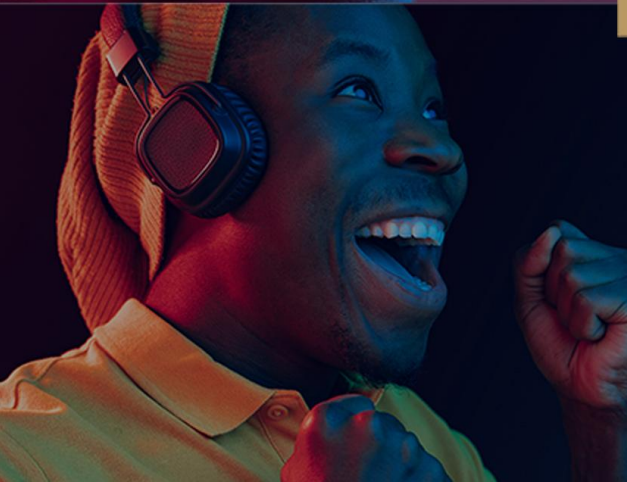
Adobe



Microsoft CompTIA.



START
YOUR TECH JOURNEY
WITH ADA



JavaScript Session 5 - Student Task

Academy of Digital Arts Egypt - JS Session 5 Task

Objective:

By completing this task, you will:

- Create objects with properties and methods using the 'this' keyword correctly
- Convert data between JavaScript objects and JSON format with proper error handling
- Implement try/catch blocks and custom error classes for robust applications
- Organize code using JavaScript modules with import/export or require syntax
- Build a complete student management system combining all learned concepts

JavaScript Session 5 - Student Task

Academy of Digital Arts Egypt - JS Session 5 Task

Step 1 – Create Student Object with Methods

- Build a Student class with properties: id, name, email, grades array, isActive
- Add methods: addGrade(), getAverage(), getInfo() using 'this' keyword
- Implement method chaining by returning 'this' from methods
- Create a updateEmail() method with validation

JavaScript Session 5 - Student Task

Academy of Digital Arts Egypt - JS Session 5 Task

Step 2 – Implement JSON Operations with Error Handling

- Add toJSON() method to convert student object to JSON string
- Create fromJSON() static method to parse JSON back to Student object
- Use try/catch blocks to handle invalid JSON parsing
- Build exportStudents() and importStudents() functions with error handling

JavaScript Session 5 - Student Task

Academy of Digital Arts Egypt - JS Session 5 Task

Step 3 – Add Custom Error Handling

- Create ValidationError class extending Error for data validation
- Implement try/catch/finally blocks in student operations
- Add error handling for: invalid email format, negative grades, missing required fields
- Create safe wrapper functions that return success/error objects

JavaScript Session 5 - Student Task

Academy of Digital Arts Egypt - JS Session 5 Task

Step 4 – Organize Code with Modules

- Split code into separate files: student.js, studentManager.js, main.js
- Use export/import (ES6) or module.exports/require (CommonJS)
- Export classes, functions, and constants between modules
- Import and use modules in main application file




JavaScript Session 5 - Student Task

Academy of Digital Arts Egypt - JS Session 5 Task

Requirements

Tools:

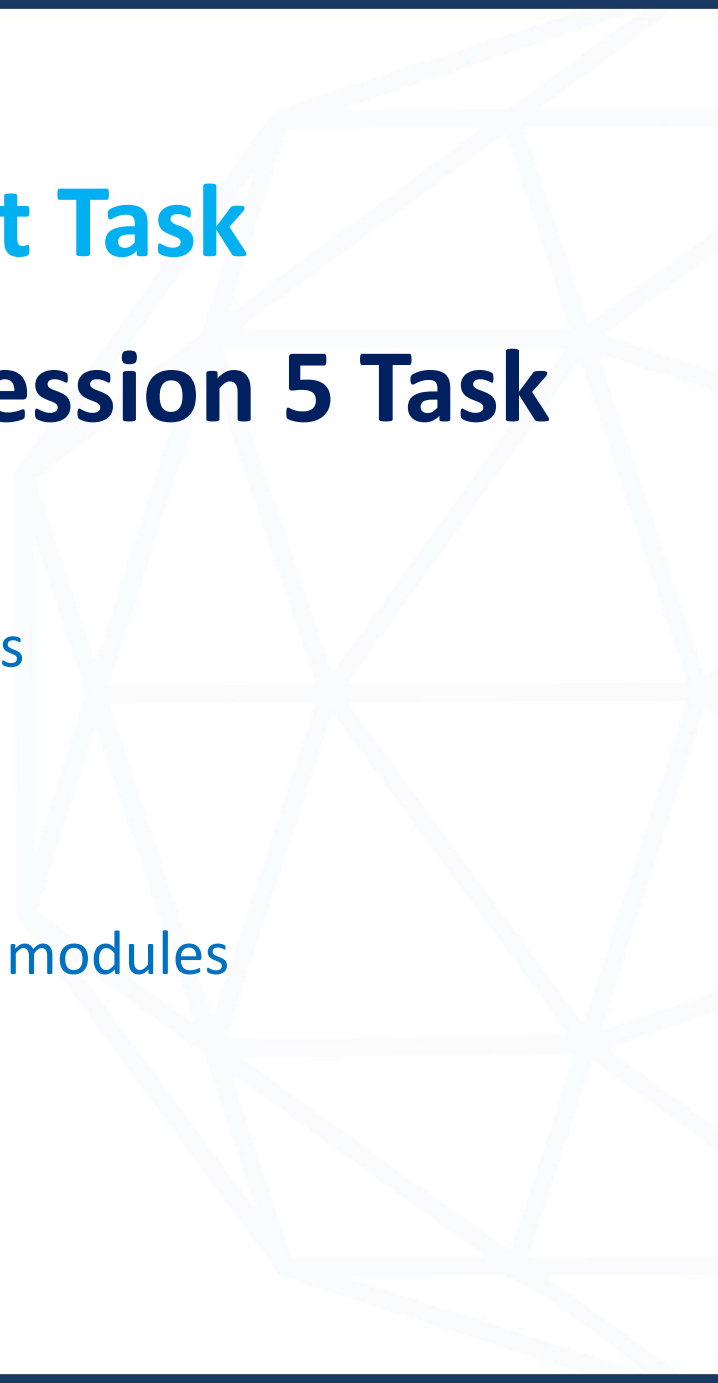
- Text editor (VS Code recommended)
 - Node.js installed for running JavaScript
 - Web browser with developer console (alternative)
- 



JavaScript Session 5 - Student Task

Academy of Digital Arts Egypt - JS Session 5 Task

Reminder

- Use 'this' keyword to access object properties within methods
 - Always wrap JSON.parse() in try/catch to handle invalid JSON
 - Create custom error classes by extending the base Error class
 - Use finally blocks for cleanup code that must always run
 - Export classes and functions to make them available in other modules
 - Test your error handling by intentionally passing invalid data
- 

THANK YOU

ADAEGY

