

KK14 — Design tools for generating solution designs from design ideas

Analysis & Design · Comprehensive exam study guide

STUDY DESIGN — YOU MUST KNOW

- data dictionaries
- mock-ups
- object descriptions
- input-process-output (IPO) charts
- pseudocode

These are the same five tools as Programming KK03, here used to turn **design ideas into solution designs**. Know each tool's purpose and be able to produce/read them.

Tool → what it designs

Tool	Designs
Data dictionary	The data: each field's name, type, size, validation.
Mock-up	The interface layout and appearance.
Object description	An object/class: its attributes and methods.
IPO chart	A module's logic as Input → Process → Output.
Pseudocode	The algorithm/steps, language-independent.

Choosing tools together

A complete design usually combines tools: mock-ups for screens, a data dictionary for stored data, and pseudocode/IPO for the processing logic. Together they let a different developer build the solution as intended.

Key definitions to memorise

Data dictionary	A table documenting each data element's name, type, size/format and validation.
Mock-up	A visual draft of an interface showing layout and controls.
Pseudocode	A language-independent, step-by-step description of an algorithm.

Exam technique

EXAM TIP

- Pick the tool by what is being designed: data → dictionary; screen → mock-up; logic → pseudocode/IPO; object → object description.
- A strong design answer often names two complementary tools.

COMMON MISTAKES

- Using real code instead of pseudocode.
- Putting interface details in a data dictionary.

Quick revision checklist

- I can match each design tool to what it designs.
- I can produce a mock-up, data-dictionary entry, IPO chart and pseudocode for a scenario.

Exam practice — questions & model answers

Q1. Identify the most suitable design tool to plan the appearance of a login screen, and one to plan its validation rules.

Identify · 2 marks

Model answer

- Appearance: a mock-up (1).
- Validation rules: a data dictionary (1).

Q2. Write pseudocode for a module that keeps asking for a password until a non-empty value is entered.

Write · 3 marks

Model answer

- INPUT password (1)
- WHILE password is empty: OUTPUT "Required"; INPUT password (1)
- ENDWHILE (1).