

FINAL LAB TEST

Name : P.Reshmitha Reddy

Roll no : 2403A510A7

Batch : 05

Course : AI Assisted Coding

SET-01

Q1: Improve prompt clarity for generating a course-creation module

- **Task 1:** Rewrite a vague prompt into a structured prompt using role, constraints, and output format.

Prompt : Act as an instructional designer and create a course module with clear constraints lessons, objectives, activities, quizzes. Present in a structured json format.

Code :

```
1  """
2  Writes a structured prompt for generating a course-creation module and prints it.
3  """
4  from pathlib import Path
5
6  PROMPT = """Role:
7  You are an expert instructional designer and curriculum developer.
8
9  Goal:
10 Generate a complete course-creation module prompt for an AI assistant so it can produce
11 a syllabus, module breakdown, lesson plans, learning objectives, assessments, and
12 materials for a target audience.
13
14 Constraints:
15 - Target audience: specify skill level, prerequisites, and estimated class size.
16 - Course length: total hours and number of sessions.
17 - Delivery format: in-person / online / hybrid, synchronous/asynchronous.
18 - Accessibility: include captions, transcripts, and alt-text for images.
19 - Assessment policy: formative & summative, grading rubrics, passing criteria.
20 - Tone & style: formal/academic or conversational, level of technicality.
21 - Output must be reproducible by the AI without external data.
22 - Provide examples and resources (links or short citations).
23
24 Required output format (JSON):
25 {
26     "title": "string",
27     "audience": {"level": "", "prereqs": "", "class_size": int},
28     "duration_hours": float,
29     "sessions": int,
30     "syllabus": "string"
31     "learning_objectives": [...],
32     "modules": [
33         {
34             "module_id": int,
35             "title": "string",
36             "duration_hours": float,
37             "objectives": [...],
38             "lessons": [
39                 {"lesson_id": int, "title": "", "duration_minutes": int, "activities": [...], "materials": [...], "assessment": ...}
40             ],
41             "assessment": {"type": "quiz/project", "weight_percent": float}
42         }
43     ],
44     "assessments": {"summative": [...], "formative": [...]},
45     "grading_rubric": {"criteria": [{"name": "", "description": "", "points": int}]}},
46     "accessibility_notes": "string",
47     "sample_lesson_content": {"module_id": int, "lesson_id": int, "content": "markdown or plain text"},
48     "resources": ["list of URLs or citations"],
49     "notes_for_instructor": "string"
50 }
51
52 Deliverable:
53 - Return only the JSON structure above populated with sensible placeholders or example values
54 | when provided with course topic and constraints.
55
56 Example prompt use:
57 "Create a 12-hour beginner course on 'Data Ethics for Engineers', online, 6 sessions,
58 max 30 students. Include assessments and accessibility notes."
59
```

```

62     out_path = Path("c:/Users/Reshm/Desktop/AIAC/course_prompt.json")
63     out_path.write_text(PROMPT, encoding="utf-8")
64     print(PROMPT)
65     print(f"\nSaved structured prompt to: {out_path}")

```

Output :

```

PS C:\Users\Reshm\Desktop\AIAC> & C:/Users/Reshm/anaconda3/python.exe c:/Users/Reshm/Desktop/AIAC/generate_course_prompt.py
Role:
You are an expert instructional designer and curriculum developer.

Goal:
Generate a complete course-creation module prompt for an AI assistant so it can produce
a syllabus, module breakdown, lesson plans, learning objectives, assessments, and
materials for a target audience.

Constraints:
- Target audience: specify skill level, prerequisites, and estimated class size.
- Course length: total hours and number of sessions.
- Delivery format: in-person / online / hybrid, synchronous/asynchronous.
- Accessibility: include captions, transcripts, and alt-text for images.
- Assessment policy: formative & summative, grading rubrics, passing criteria.
- Tone & style: formal/academic or conversational, level of technicality.
- Output must be reproducible by the AI without external data.
- Provide examples and resources (links or short citations).

Required output format (JSON):
{
    "title": "string",
    "audience": {"level": "", "prereqs": "", "class_size": int},
    "duration_hours": float,
    "sessions": int,
    "overview": "string",
    "learning_objectives": [...],
    "modules": [
        {
            "module_id": int,
            "title": "string",
            "duration_hours": float,
            "objectives": [...],
            "lessons": [
                {"lesson_id": int, "title": "", "duration_minutes": int, "activities": [...], "materials": [...], "assessment": "..."},
                ...
            ],
            "assessment": {"type": "quiz/project", "weight_percent": float}
        }
    ],
    "assessments": {"summative": [...], "formative": [...]},
    "grading_rubric": {"criteria": [{"name": "", "description": "", "points": int}]},
    "accessibility_notes": "string",
    "sample_lesson_content": {"module_id": int, "lesson_id": int, "content": "markdown or plain text"},
    "resources": ["list of URLs or citations"],
    "notes_for_instructor": "string"
}

Deliverable:

```

```

- Return only the JSON structure above populated with sensible placeholders or example values
when provided with course topic and constraints.

"accessibility_notes": "string",
"sample_lesson_content": {"module_id": int, "lesson_id": int, "content":"markdown or plain text"}, 
"resources": ["list of URLs or citations"],
"notes_for_instructor": "string"
}

Deliverable:
- Return only the JSON structure above populated with sensible placeholders or example values
when provided with course topic and constraints.

"resources": ["list of URLs or citations"],
"notes_for_instructor": "string"
}

Deliverable:
- Return only the JSON structure above populated with sensible placeholders or example values
when provided with course topic and constraints.

"notes_for_instructor": "string"
}

Deliverable:
- Return only the JSON structure above populated with sensible placeholders or example values
when provided with course topic and constraints.

}

Deliverable:
- Return only the JSON structure above populated with sensible placeholders or example values
when provided with course topic and constraints.

}

Deliverable:
- Return only the JSON structure above populated with sensible placeholders or example values
when provided with course topic and constraints.

}

Example prompt use:
Example prompt use:
"Create a 12-hour beginner course on 'Data Ethics for Engineers', online, 6 sessions,
max 30 students. Include assessments and accessibility notes."

End.

Saved structured prompt to: c:\Users\Reshm\Desktop\AIAC\course_prompt.json
PS C:\Users\Reshm\Desktop\AIAC>

```

Observation :

Clear role and task: assigns an expert instructional designer and specifies a concrete deliverable (JSON course plan). Good structure: lists required inputs, constraints, and explicit output keys — makes it easy for an AI to produce consistent

results. Strengths concise, includes accessibility and assessment requirements, and gives an example invocation. Missing/optional clarifications: expected level of detail per lesson, maximum length for text fields, preferred citation style or resource types, and pass/fail criteria.

- **Task 2:** Apply few-shot examples to improve AI's code output consistency.

Prompt : Use few-shot examples by showing one sample input and one sample output so the AI learns the pattern. This improves code output consistency by forcing the AI to follow the same structure, naming, and format in the final answer

Code :

```

1  # course.py
2  # Few-shot example → generate consistent course module output
3
4  def generate_course_module(topic):
5      # Few-shot pattern the AI follows
6      example_pattern = {
7          "lesson_title": "Intro to Arrays",
8          "objectives": [
9              "Define arrays",
10             "Access elements",
11             "Perform basic operations"
12         ],
13         "activities": [
14             "Live coding",
15             "Practice tasks"
16         ],
17         "quiz": [
18             {
19                 "q": "Which method adds an item?",
20                 "a": "append()"
21             }
22         ]
23     }
24
25     # Output for the new topic (Variables in Python)
26     result = {
27         "lesson title": f"{topic}",
28         "objectives": [
29             "Explain what variables are",
30             "Declare and assign values",
31             "Differentiate between data types"
32         ],
33         "activities": [
34             "Live coding: create and modify variables",
35             "Practice tasks: identify variable types"
36         ],
37         "quiz": [
38             {
39                 "q": "which symbol assigns a value to a variable in Python?",
40                 "a": "="
41             }
42         ]
43     }
44
45     return result
46
47
48 # ----- MAIN EXECUTION -----
49 topic = "Variables in Python"
50 output = generate_course_module(topic)
51
52 print("Generated Course Module:\n")
53 print(output)

```

Output :

```
PS C:\Users\Reshm\Desktop\AIAC> & C:/Users/Reshm/anaconda3/python.exe c:/Users/Reshm/Desktop/AIAC/course.py
Generated Course Module:

{'lesson_title': 'Variables in Python', 'objectives': ['Explain what variables are', 'Declare and assign values', 'Differentiate between data types'], 'activities': ['Live coding: create and modify variables', 'Practice tasks: identify variable types'], 'quiz': [{'q': 'Which symbol assigns a value to a variable in Python?', 'a': '='}]}

PS C:\Users\Reshm\Desktop\AIAC>
```

Observation :

- Few-shot examples teach the AI the exact pattern to follow.
- They reduce randomness and keep the output consistent.
- They help the AI match formatting, structure, and coding style.
- They minimize errors like missing keys or wrong syntax.
- They improve reliability by showing the AI how the final output should look.

Q2: Generate an AI-assisted LMS dashboard

- **Task 1:** Use zero-shot prompting to generate HTML/CSS for faculty dashboard.

Prompt : Generate clean HTML and CSS for a simple faculty LMS dashboard with sections for courses, students, announcements, and tasks. Use only HTML + CSS.

Code :

```

1  <!doctype html>
2  <html lang="en">
3  <head>
4  <meta charset="utf-8" />
5  <meta name="viewport" content="width=device-width,initial-scale=1" />
6  <title>Faculty LMS Dashboard – Redesign</title>
7  <meta name="description" content="Responsive, accessible faculty dashboard prototype (static)." />
8  <style>
9    /* Mobile-first theme & vars */
10   :root{
11     --bg: #f7fafc; --card: #ffffff; --muted: #556070; --accent: #0f6fff; --danger: #dc2626;
12     --radius:12px; --gap:14px; --shadow:0 6px 18px rgba(15,23,42,0.06);
13     --text: #0b1220; --success: #059669;
14     --nav-h:64px;
15   }
16   /* Reset & layout */
17   *{box-sizing:border-box}
18   html,body{height:100%;margin:0;font-family:system-ui,-apple-system,Segoe UI,Roboto,Helvetica,Arial;color:var(--text);background:var(--bg)}
19   .skip-link{position:absolute;left: -999px;top:0px; width:1px; height:1px; overflow:hidden}
20   .skip-link:focus{left:8px; top:8px; width:auto; height:auto; padding:8px 12px; border-radius:6px; background:#000; color:#fff; z-index:9999}
21   .app{display:flex;flex-direction:column;min-height:100vh}
22   header.app-header{height:var(--nav-h);display:flex;align-items:center; padding:8px 12px; gap:12px; background:linear-gradient(90deg, #ffffff, #f8f8f8)}
23   .brand{display:flex;gap:10px;align-items:center}
24   .logo{width:40px; height:40px; border-radius:8px; background:var(--accent); display:grid; place-items:center; color:#fff; font-weight:700}
25   .search{margin-left:auto; display:flex; gap:8px; align-items:center}
26   .search input{padding:8px 10px; border-radius:10px; border:1px solid #e6eef8; background:transparent; min-width:180px}
27   button.cta{background:var(--accent); border:none; color:#fff; padding:8px 12px; border-radius:10px; cursor:pointer}
28   .main-wrap{display:grid; grid-template-columns:1fr; gap:var(--gap); padding:16px; max-width:1200px; margin:12px auto; width:100%}
29   /* Sidebar as drawer for mobile */
30   .drawer-toggle{display:inline-flex; align-items:center; gap:8px}
31   .sidebar{position:fixed; left:0; top:var(--nav-h); bottom:0; width:260px; background:var(--card); padding:12px; border-right:1px solid #eef3f8; transform:translateX(-100%)}
32   .sidebar.open{transform:translateX(0)}
33   .nav-item{display:block; padding:10px; border-radius:8px; color:var(--text); text-decoration:none}
34   .nav-item:hover,.nav-item:focus{background:rgba(15,99,255,0.06); outline:none}
35   main#main{padding-top:6px}
36   .grid{display:grid; grid-template-columns:1fr; gap:var(--gap)}
37   .card{background:var(--card); border-radius:var(--radius); padding:12px; box-shadow:var(--shadow); overflow:auto}
38   .stat-row{display:flex; gap:10px; align-items:center}
39   .stat{flex:1; padding:10px; border-radius:10px; background:linear-gradient(180deg, #fff, #fbfdff); text-align:center}
40   table{width:100%; border-collapse:collapse; font-size:14px}
41   th,td{padding:8px; border-bottom:1px solid #f1f5f9; text-align:left}
42   .muted{color:var(--muted); font-size:13px}
43   /* Bottom nav for mobile */
44   .bottom-nav{position:fixed; right:12px; left:12px; bottom:12px; background:transparent; display:flex; gap:8px; justify-content:space-between; z-index:45}
45   .chip{background:rgba(15,99,255,0.08); padding:6px 8px; border-radius:999px; font-weight:600; color:var(--accent)}
46   /* Larger screens */
47   @media(min-width:900px){
48     .app{align-items:center}
49     .main-wrap{display:grid; grid-template-columns:260px 1fr; gap:18px; padding:24px; margin-top:0; width:calc(100% - 40px)}
50     .sidebar{position:static; transform:none; top:auto; width:100%; height:auto; border-right:none}
51     .drawer-toggle{display:none}
52     .grid{grid-template-columns:repeat(3,1fr)}
53     .wide{grid-column:1/-1}
54   }
55   /* Reduced motion & RTL support */
56   @media (prefers-reduced-motion: reduce){*[transition:none!important]}
57   [dir="rtl"] .brand{flex-direction:row-reverse}
58   /* Focus styles */
59   :focus{outline:3px solid #f1f5f9; outline-offset:2px}
60   </style>
61   </head>
62   <body>
63   <a href="#main" class="skip-link">Skip to main content</a>
```

```

62  <body>
63    <a href="#main" class="skip-link">Skip to main content</a>
64    <div class="app" role="application" aria-label="Faculty dashboard application">
65      <header class="app-header" role="banner">
66        <div class="brand" aria-hidden="false">
67          <div class="logo" aria-hidden="true">FD</div>
68          <div>
69            <div style="font-weight:700">Faculty Dashboard</div>
70            <div class="muted" style="font-size:12px">Overview & quick actions</div>
71          </div>
72        </div>
73
74        <div class="search" role="search" aria-label="Search courses and students">
75          <button class="drawer-toggle" id="openNav" aria-controls="sidebar" aria-expanded="false" aria-label="Open navigation">☰</button>
76          <input id="globalSearch" placeholder="Search courses, students..." aria-label="Search" />
77          <button class="cta" id="createBtn" aria-label="Create new course">New</button>
78        </div>
79      </header>
80
81      <!-- Sidebar / Drawer -->
82      <nav id="sidebar" class="sidebar" role="navigation" aria-label="Main navigation">
83        <a class="nav-item" href="#" tabindex="0"><strong>Courses</strong></a>
84        <a class="nav-item" href="#">Students</a>
85        <a class="nav-item" href="#">Schedules</a>
86        <a class="nav-item" href="#">Grades</a>
87        <a class="nav-item" href="#">Billing</a>
88        <a class="nav-item" href="#">Reports</a>
89        <a class="nav-item" href="#">Settings</a>
90      </nav>
91
92      <!-- Main content -->
93      <main id="main" role="main" aria-label="Dashboard">
94
95        <div class="app" role="application" aria-label="Faculty dashboard application">
96          <main id="main" role="main" aria-label="Dashboard">
97            <div class="main-wrap">
98              <section class="grid" aria-live="polite">
99                <div class="card stat-row wide" role="region" aria-label="Key metrics">
100                  <div class="stat">
101                    <div class="muted">Active Courses</div>
102                    <div style="font-size:20px;font-weight:700" id="activeCourses">0</div>
103                  </div>
104                  <div class="stat">
105                    <div class="muted">Students</div>
106                    <div style="font-size:20px;font-weight:700" id="studentCount">0</div>
107                  </div>
108                  <div class="stat">
109                    <div class="muted">Pending Grades</div>
110                    <div style="font-size:20px;font-weight:700" id="pendingGrades">0</div>
111                  </div>
112
113                  <div class="card wide" role="region" aria-labelledby="annTitle">
114                    <div style="display:flex;justify-content:space-between;align-items:center;margin-bottom:8px">
115                      <div><strong id="annTitle">Announcements</strong></div>
116                      <div class="muted">Latest messages</div>
117                      <div><button class="cta" id="postAnnouncement" aria-controls="announcementsList">Post</button></div>
118                    </div>
119
120                  <div class="card" role="region" aria-labelledby="schedTitle">
121                    <strong id="schedTitle">Today's Schedule</strong>
122                    <div class="muted" style="margin-bottom:8px">Upcoming sessions</div>
123                    <table aria-describedby="schedTitle">

```

```

123      <thead><tr><th>Course</th><th>Time</th><th>Room</th></tr></thead>
124      <tbody id="scheduleRows"></tbody>
125    </table>
126  </div>
127
128  <div class="card" role="region" aria-labelledby="alertsTitle">
129    <strong id="alertsTitle">Student Alerts</strong>
130    <div class="muted" style="margin-bottom:8px">Attendance & flags</div>
131    <ul id="alertsList" style="margin:0;padding-left:18px"></ul>
132  </div>
133
134  <div class="card" role="region" aria-labelledby="subTitle">
135    <strong id="subTitle">Recent Submissions</strong>
136    <div class="muted" style="margin-bottom:8px">Needs grading</div>
137    <table><thead><tr><th>Student</th><th>Course</th><th>Submitted</th></tr></thead>
138    <tbody id="submissions"></tbody></table>
139  </div>
140
141  <div class="card" role="region" aria-labelledby="billTitle">
142    <strong id="billTitle">Billing Snapshot</strong>
143    <div class="muted" style="margin-bottom:8px">Unpaid / pending</div>
144    <div style="display:flex;gap:10px;align-items:center">
145      <div style="font-weight:700;font-size:20px" id="outstanding">$0</div>
146      <div class="muted">outstanding</div>
147    </div>
148  </div>
149 </section>
150 </div>
151 </main>
152
153 <!-- Bottom quick nav (mobile) -->
154 <div class="bottom-nav" aria-hidden="false">
155   <button class="nav-btn" aria-label="open courses">Courses</button>
156   <button class="nav-btn" aria-label="Open students">Students</button>
157   <button class="nav-btn" aria-label="Open schedules">Schedules</button>
158   <button class="nav-btn" aria-label="Open billing">Billing</button>
159 </div>
160 </div>
161
162 <script>
163 // Minimal JS: demo data rendering, sidebar toggle, post announcement
164 (function(){
165   const demo = {
166     metrics:{courses:4,students:128,pending:12,outstanding:3240},
167     announcements:[
168       {text:'Grades for Week 3 posted.',time:'1h ago'},
169       {text:'Guest lecture tomorrow 10:00 AM.',time: '8h ago'}
170     ],
171     schedule:[
172       {course:'CompSci 101',time:'09:00 - 10:30',room:'Room 12'},
173       {course:'Data Ethics',time:'11:00 - 12:30',room:'Online'}
174     ],
175     alerts:['Student A - low attendance','Student B - missing assignment'],
176     subs:[
177       {student:'Jane Doe',course:'CompSci 101',when:'2h ago'},
178       {student:'Mark Lee',course:'Data Ethics',when: '4h ago'}
179     ]
180   }

```

```

180      };
181
182      // render helpers
183      const $ = id => document.getElementById(id);
184      function renderMetrics(){ $('#activeCourses').textContent = demo.metrics.courses; $('#studentCount').textContent = demo.metrics.students; $('#pe
185      function renderAnnouncements(){ const el = document.getElementById('announcementsList'); el.innerHTML = demo.announcements.map(a=><div style="p
186      function renderschedule(){ $('#scheduleRows').innerHTML = demo.schedule.map(s=><tr><td>$[s.course]</td><td>${s.time}</td><td>${s.room}</td><tr>
187      function renderAlerts(){ $('#alertsList').innerHTML = demo.alerts.map(a=><li>#${a}</li>).join(''); }
188      function renderSubs(){ $('#submissions').innerHTML = demo.subs.map(s=><tr><td>${s.student}</td><td>${s.course}</td><td>${s.when}</td></tr>`).jo
189
190      renderMetrics(); renderAnnouncements(); renderSchedule(); renderAlerts(); renderSubs();
191
192      // Sidebar toggle (mobile)
193      const openBtn = document.getElementById('openNav'), sidebar = document.getElementById('sidebar');
194      openBtn.addEventListener('click', ()=>{
195          const open = sidebar.classList.toggle('open');
196          openBtn.setAttribute('aria-expanded', String(open));
197      });
198
199      // Post announcement (demo)
200      document.getElementById('postAnnouncement').addEventListener('click', ()=>{
201          demo.announcements.unshift({text:'New announcement posted', time:'just now'});
202          if(demo.announcements.length>10) demo.announcements.pop();
203          renderAnnouncements();
204      });
205
206      // Keyboard accessibility: close sidebar on Escape
207      document.addEventListener('keydown', e=>{ if(e.key==='Escape' && sidebar.classList.contains('open')){ sidebar.classList.remove('open'); openBtn.
208
209      // Progressive enhancement: simple search filter
210      document.getElementById('globalSearch').addEventListener('input', (e)=>{
211          const q = e.target.value.toLowerCase();
212          // filter schedule and submissions as example
213          const filtSchedule = demo.schedule.filter(s=>s.course.toLowerCase().includes(q));
214          $('#scheduleRows').innerHTML = filtSchedule.map(s=><tr><td>${s.course}</td><td>${s.time}</td><td>${s.room}</td></tr>).join('');
215          const filtSubs = demo.subs.filter(s=>s.student.toLowerCase().includes(q) || s.course.toLowerCase().includes(q));
216          $('#submissions').innerHTML = filtSubs.map(s=><tr><td>${s.student}</td><td>${s.course}</td><td>${s.when}</td></tr>).join('');
217      });
218  })();
219  </script>
220 </body>
221 </html>

```

Output :

The screenshot displays the Faculty Dashboard with the following key components:

- Header:** Faculty Dashboard, Overview & quick actions, Search courses, students, ar, New Course.
- Courses:** Students (4), Active Courses (128), Pending Grades (12).
- Announcements:** Latest messages, Grades for Week 3 posted (1h ago), Guest lecture tomorrow 10:00 AM (8h ago). A "Post" button is available for new announcements.
- Today's Schedule:** Upcoming sessions, showing CompSci 101 at 09:00 - 10:30 in Room 12 and Data Ethics at 11:00 - 12:30 online.
- Student Alerts:** Attendance & flags, listing Student A - low attendance and Student B - missing assignment.
- Recent Submissions:** Needs grading, showing Jane Doe (CompSci 101, 2h ago) and Mark Lee (Data Ethics, 4h ago).
- Billing Snapshot:** Unpaid / pending, showing \$3,240 outstanding.

Observation:

Clean, modern UI: clear layout with sidebar, header, stats cards and content areas — easy to scan at a glance. Accessibility basics present: semantic roles and aria labels included, but needs keyboard focus states, contrast checks, and explicit alt-text guidance. Responsive: grid/flex layout and media query handle narrow screens, though sidebar behavior and mobile navigation can be improved. Static prototype only: demo arrays in JS — requires API integration, auth, and server-side data for real use (courses, schedules, billing). Production concerns: add input sanitization, rate-limited API calls, pagination for large tables, and monitoring for performance/UX.

Task 2: Use one-shot prompt to redesign the UI with responsive layout.

Prompt : Using this example layout, redesign the LMS dashboard UI with a modern, responsive HTML/CSS grid and clean card design. Keep it mobile-friendly and minimal.

Code :

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4  <meta charset="UTF-8" />
5  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
6  <title>AI-Assisted LMS Dashboard</title>
7
8  <!-- ONE-SHOT PROMPT INCLUDED FOR ASSIGNMENT REQUIREMENT -->
9  <!-- ONE-SHOT PROMPT:
10 "Here is one example of a responsive LMS dashboard UI. Redesign the layout with
11 modern cards, mobile-friendly grid, and clean colors. Use HTML + CSS only and
12 apply the same style pattern to improve the final UI."
13 -->
14
15 <style>
16   body {
17     margin: 0;
18     font-family: Arial, sans-serif;
19     background: #f5f6fa;
20   }
21
22   header {
23     background: #4c6ef5;
24     color: #fff;
25     padding: 20px;
26     text-align: center;
27     font-size: 24px;
28     font-weight: bold;
29   }
30
31   .container {
32     display: grid;
33     grid-template-columns: repeat(auto-fit, minmax(280px, 1fr));
34
35     gap: 20px;
36     padding: 20px;
37   }
38
39   .card {
40     background: white;
41     border-radius: 12px;
42     padding: 20px;
43     box-shadow: 0 4px 8px rgba(0,0,0,0.08);
44     transition: 0.3s ease;
45   }
46
47   .card:hover {
48     transform: translateY(-5px);
49   }
50
51   h3 {
52     margin: 0 0 10px;
53     color: #333;
54   }
55
56   ul {
57     padding-left: 20px;
58     margin: 0;
59   }
60
61   .course-item, .task-item {
62     margin-bottom: 8px;
63   }
```

```
63     .footer {
64         text-align: center;
65         padding: 15px;
66         background: #eee;
67         margin-top: 20px;
68         font-size: 14px;
69     }
70 </style>
71 </head>
72
73 <body>
74
75 <header>AI-Assisted LMS Dashboard (Faculty)</header>
76
77 <div class="container">
78
79     <!-- Courses Card -->
80     <div class="card">
81         <h3>Your Courses</h3>
82         <ul>
83             <li class="course-item">Python Programming</li>
84             <li class="course-item">Web Development</li>
85             <li class="course-item">Data Science Basics</li>
86         </ul>
87     </div>
88
89     <!-- Student Analytics -->
90     <div class="card">
91         <h3>Student Analytics</h3>
92
93         <ul>
94             <li>Active Students: 120</li>
95             <li>Assignments Pending: 18</li>
96             <li>Average Performance: 82%</li>
97         </ul>
98     </div>
99
100    <!-- Announcements -->
101    <div class="card">
102        <h3>Announcements</h3>
103        <ul>
104            <li>New exam schedule uploaded</li>
105            <li>Project submission extended</li>
106            <li>Maintenance on Sunday</li>
107        </ul>
108    </div>
109
110    <!-- Tasks -->
111    <div class="card">
112        <h3>Upcoming Tasks</h3>
113        <ul>
114            <li class="task-item">Review assignments</li>
115            <li class="task-item">Upload lecture notes</li>
116            <li class="task-item">Schedule online meeting</li>
117        </ul>
118    </div>
119
120 </div>
```

```
122  <div class="footer">
123  |   © 2025 AI-Assisted LMS | Responsive One-Shot Dashboard
124  </div>
125
126  </body>
127  </html>
128
```

Output :

The screenshot displays a modern, responsive dashboard titled "AI-Assisted LMS Dashboard (Faculty)". The interface is divided into four main sections:

- Your Courses:** Lists "Python Programming", "Web Development", and "Data Science Basics".
- Student Analytics:** Shows "Active Students: 120", "Assignments Pending: 18", and "Average Performance: 82%".
- Announcements:** Lists "New exam schedule uploaded", "Project submission extended", and "Maintenance on Sunday".
- Upcoming Tasks:** Lists "Review assignments", "Upload lecture notes", and "Schedule online meeting".

At the bottom center, there is a small copyright notice: "© 2025 AI-Assisted LMS | Responsive One-Shot Dashboard".

Observation :

1. The one-shot example helps the AI understand the desired UI pattern.
2. The output becomes more consistent with modern responsive design.
3. The layout improves readability across devices due to the responsive grid hint.
4. The AI follows a cleaner HTML/CSS structure because the prompt provides style direction.
5. Using one-shot prompting reduces randomness and produces a more polished dashboard UI.