

React Props Assignment

Project Title:

AI-Enabled Team Insights Panel (Props-Driven UI)

Objective:

Assess your capability to build scalable, component-based UIs using **React props as the sole data flow mechanism**, aligned with real-world SaaS and AI dashboard requirements.

Project Overview

Your team is building the early UI for an internal **AI Workforce Intelligence Platform**.

The engineering team wants a **Props-only prototype** to validate data-flow behavior before connecting it to backend and ML models.

You are required to create a **team insights panel** that displays:

- basic member info
- AI-generated metrics
- visual tags
- status indicators

All of this **must be passed via props only**.



Requirements

1. Parent Component: TeamDashboard

Must include:

- A `team` array with **5 members**, each containing:
 - `name`
 - `role`
 - `experience`
 - `skills` (array)
 - `aiScore` (0–100)
 - `status` (“active”, “bench”, “interviewing”)

Responsibilities:

- Pass each member object into `<TeamCard />` via props.
 - No transformation inside child — all pre-processing happens in parent.
-

2. Child Component: `TeamCard`

Must:

- Receive all values **strictly via props**.
- Use destructuring.
- Display:
 - Name & role
 - Experience
 - Status pill (color varies via prop value)
 - AI Score
 - Skill tags

- Should not hold state, effects, or computed logic.
-

3. UI Expectations

- Clean **SaaS-style layout** (basic styling OK).
 - Consistent spacing & readable typography.
 - Status indicator styling must be purely **prop-driven**.
 - No conditional logic inside children except *rendering UI based on props*.
-



Additional Functional Expectations

A. Reusable Tag Component

Create a **Tag** component that receives:

- `label`
- `type` (“skill”, “ai”, etc.)

Used inside `<TeamCard />` for skills.

B. AI Score Highlight

If `aiScore > 85`, visually highlight:

- either border
- or background
- or subtle glow



Folder Structure (Required)

```
src/
  components/
    TeamDashboard.jsx
    TeamCard.jsx
    Tag.jsx
  data/
    teamData.js  (Optional: If you want to separate data)
App.jsx
```



Deliverables

- All component files
- Clean, readable JSX
- No state, context, or external libs
- Only props for data transfer
- Working render of all 5 members