Task 5

1.Task Description:

Create a shared input component that can handle multiple types of inputs, such as text, textarea, checkbox, and radio. This component standardizes form handling across the app.

2.Task Output Screenshot

Shared Input Component

Text Input:

Enter textInput

Textarea Input:

Enter your text here

Checkbox Input:
☐ Option 1   ☐ Option 2

Radio Input:
○ Yes   ○ No

Form Data

```
{
  "textInput": "",
  "textareaInput": "",
  "checkboxInput": [],
  "radioInput": ""
}
```

Shared Input Component

Text Input:

Yash Mehta

Textarea Input:

djfskfbsmnbskvshflfns,vndmnbdksfns,

Checkbox Input:
☑ Option 1 ☐ Option 2

Radio Input:
○ Yes ◉ No

Form Data

```
{
  "textInput": "Yash Mehta",
  "textareaInput": "djfskfbsmnbskvshflfns,vndmnbdksfns,",
  "checkboxInput": [
    "option1"
  ],
  "radioInput": "no"
}
```

3.Widget/Algorithm Used In Task

a. **Reusable Component Design:** The input component can accept props that define its type (e.g., text, checkbox) and render accordingly.

b. Dynamic Rendering: Renders different input types based on props, providing flexibility and reducing repetitive code.

c. Event Handling: Centralized onChange event to capture data input, allowing each input type to trigger specific behaviors if needed.

d. useState Hook: Keeps track of input values, providing a simple state management solution for form elements.