

```

// CustomButton.js
import React from "react";
import PropTypes from "prop-types";
import { Button } from "antd";
import { IconContext } from "react-icons";

const SlipButton = ({ icon, text, size, type, onClick, style, disabled }) => {
  const buttonType = type === "outline" ? "ghost" : type;
  const getCustomClass = (size) => {
    switch (size) {
      case "small":
        return "custom-small";
      case "middle":
        return "custom-middle";
      case "large":
        return "custom-large";
      default:
        return "";
    }
  };
  const getCustomOutlineClass = (size) => {
    switch (size) {
      case "small":
        return "custom-outline-small";
      case "middle":
        return "custom-outline-middle";
      case "large":
        return "custom-outline-large";
      default:
        return "";
    }
  };
  return (
    <Button
      disabled={disabled}
      style={style}
      type={buttonType}
      size={size}
      onClick={onClick}
      className={
        type === "outline" ? getCustomOutlineClass(size) : getCustomClass(size)
      }
      icon={
        icon && (
          <IconContext.Provider value={{ size: "1em", className: "mr-2" }}>
            {React.createElement(icon)}
          </IconContext.Provider>
        )
      }
    >
      {text}
    </Button>
  );
};

SlipButton.propTypes = {
  icon: PropTypes.elementType,
  text: PropTypes.string.isRequired,
  size: PropTypes.oneOf(["small", "middle", "large"]),
  type: PropTypes.oneOf([
    "primary",
    "ghost",
    "dashed",
    "link",
    "text",
    "default",
    "outline",
  ]),
  onClick: PropTypes.func,
};

```

```
export default SlipButton;
```