

File: client/src/components/ui/calendar.tsx

Type: tsx | Size: 2695 bytes

```
import * as React from "react"
import { ChevronLeft, ChevronRight } from "lucide-react"
import { DayPicker } from "react-day-picker"

import { cn } from "@/lib/utils"
import { buttonVariants } from "@/components/ui/button"

export type CalendarProps = React.ComponentProps<typeof DayPicker>

function Calendar({
  className,
  classNames,
  showOutsideDays = true,
  ...props
}: CalendarProps) {
  return (
    <DayPicker
      showOutsideDays={showOutsideDays}
      className={cn("p-3", className)}
      classNames={{
        months: "flex flex-col sm:flex-row space-y-4 sm:space-x-4 sm:space-y-0",
        month: "space-y-4",
        caption: "flex justify-center pt-1 relative items-center",
        caption_label: "text-sm font-medium",
        nav: "space-x-1 flex items-center",
        nav_button: cn(
          buttonVariants({ variant: "outline" }),
          "h-7 w-7 bg-transparent p-0 opacity-50 hover:opacity-100"
        ),
        nav_button_previous: "absolute left-1",
        nav_button_next: "absolute right-1",
        table: "w-full border-collapse space-y-1",
        head_row: "flex",
        head_cell: "text-muted-foreground rounded-md w-9 font-normal text-[0.8rem]",
        row: "flex w-full mt-2",
        cell: "h-9 w-9 text-center text-sm p-0 relative [&:has([aria-selected].day-range-end)]:rounded-r-md [&...",
        day: cn(
          buttonVariants({ variant: "ghost" }),
          "h-9 w-9 p-0 font-normal aria-selected:opacity-100"
        ),
        day_range_end: "day-range-end",
        day_selected: "bg-primary text-primary-foreground hover:bg-primary hover:text-primary-foreground focus:bg-primary ...",
        day_today: "bg-accent text-accent-foreground",
        day_outside: "day-outside text-muted-foreground aria-selected:bg-accent/50 aria-selected:text-muted-foreground",
        day_disabled: "text-muted-foreground opacity-50",
        day_range_middle: "aria-selected:bg-accent aria-selected:text-accent-foreground",
        day_hidden: "invisible",
        ...classNames,
      })
      components={{
        IconLeft: ({ className, ...props }) => (
          <ChevronLeft className={cn("h-4 w-4", className)} {...props} />
        ),
        IconRight: ({ className, ...props }) => (
          <ChevronRight className={cn("h-4 w-4", className)} {...props} />
        ),
      }}
      {...props}
    />
  )
}

Calendar.displayName = "Calendar"

export { Calendar }
```