* /Client/public/
* /android-chrome-192x192.png

/android-chrome-512x512.png

* /apple-touch-icon.png
* /favicon.ico
* /favicon.svg
* /logo.jpeg
* /client
* /public
* logo.jpeg/



/Src/Components/ui/button.tsx

import \* as React from 'react';

import { Slot } from '@radix-ui/react-slot';

import { cva, type VariantProps } from 'class-variance-authority';

import { cn } from '@/lib/utils';

const buttonVariants = cva(

'inline-flex items-center justify-center gap-2 whitespace-nowrap rounded-md text-sm font-medium transition-colors focus-visible:outline-none focus-visible:ring-1 focus-visible:ring-ring disabled:pointer-events-none disabled:opacity-50 [&\_svg]:pointer-events-none [&\_svg]:size-4 [&\_svg]:shrink-0',

{

variants: {

variant: {

default:

'bg-primary text-primary-foreground shadow hover:bg-primary/90',

destructive:

'bg-destructive text-destructive-foreground shadow-sm hover:bg-destructive/90',

outline:

'border border-input bg-background shadow-sm hover:bg-accent hover:text-accent-foreground',

secondary:

'bg-secondary text-secondary-foreground shadow-sm hover:bg-secondary/80',

ghost: 'hover:bg-accent hover:text-accent-foreground',

link: 'text-primary underline-offset-4 hover:underline',

},

size: {

default: 'h-9 px-4 py-2',

sm: 'h-8 rounded-md px-3 text-xs',

lg: 'h-10 rounded-md px-8',

icon: 'h-9 w-9',

},

},

defaultVariants: {

variant: 'default',

size: 'default',

},

},

);

export interface ButtonProps

extends React.ButtonHTMLAttributes<HTMLButtonElement>,

VariantProps<typeof buttonVariants> {

asChild?: boolean;

}

const Button = React.forwardRef<HTMLButtonElement, ButtonProps>(

({ className, variant, size, asChild = false, ...props }, ref) => {

const Comp = asChild ? Slot : 'button';

return (

<Comp

className={cn(buttonVariants({ variant, size, className }))}

ref={ref}

{...props}

/>

);

},

);

Button.displayName = 'Button';

export { Button, buttonVariants };

Calendar.tsx

import \* as React from 'react';

import {

ChevronDownIcon,

ChevronLeftIcon,

ChevronRightIcon,

} from 'lucide-react';

import { DayButton, DayPicker, getDefaultClassNames } from 'react-day-picker';

import { cn } from '@/lib/utils';

import { Button, buttonVariants } from '@/components/ui/button';

function Calendar({

className,

classNames,

showOutsideDays = true,

captionLayout = 'label',

buttonVariant = 'ghost',

formatters,

components,

...props

}: React.ComponentProps<typeof DayPicker> & {

buttonVariant?: React.ComponentProps<typeof Button>['variant'];

}) {

const defaultClassNames = getDefaultClassNames();

return (

<DayPicker

showOutsideDays={showOutsideDays}

className={cn(

'bg-background group/calendar p-3 [--cell-size:2rem] [[data-slot=card-content]\_&]:bg-transparent [[data-slot=popover-content]\_&]:bg-transparent',

String.raw`rtl:\*\*:[.rdp-button\\_next>svg]:rotate-180`,

String.raw`rtl:\*\*:[.rdp-button\\_previous>svg]:rotate-180`,

className,

)}

captionLayout={captionLayout}

formatters={{

formatMonthDropdown: (date) =>

date.toLocaleString('default', { month: 'short' }),

...formatters,

}}

classNames={{

root: cn('w-fit', defaultClassNames.root),

months: cn(

'relative flex flex-col gap-4 md:flex-row',

defaultClassNames.months,

),

month: cn('flex w-full flex-col gap-4', defaultClassNames.month),

nav: cn(

'absolute inset-x-0 top-0 flex w-full items-center justify-between gap-1',

defaultClassNames.nav,

),

button\_previous: cn(

buttonVariants({ variant: buttonVariant }),

'h-[--cell-size] w-[--cell-size] select-none p-0 aria-disabled:opacity-50',

defaultClassNames.button\_previous,

),

button\_next: cn(

buttonVariants({ variant: buttonVariant }),

'h-[--cell-size] w-[--cell-size] select-none p-0 aria-disabled:opacity-50',

defaultClassNames.button\_next,

),

month\_caption: cn(

'flex h-[--cell-size] w-full items-center justify-center px-[--cell-size]',

defaultClassNames.month\_caption,

),

dropdowns: cn(

'flex h-[--cell-size] w-full items-center justify-center gap-1.5 text-sm font-medium',

defaultClassNames.dropdowns,

),

dropdown\_root: cn(

'has-focus:border-ring border-input shadow-xs has-focus:ring-ring/50 has-focus:ring-[3px] relative rounded-md border',

defaultClassNames.dropdown\_root,

),

dropdown: cn('absolute inset-0 opacity-0', defaultClassNames.dropdown),

caption\_label: cn(

'select-none font-medium',

captionLayout === 'label'

? 'text-sm'

: '[&>svg]:text-muted-foreground flex h-8 items-center gap-1 rounded-md pl-2 pr-1 text-sm [&>svg]:size-3.5',

defaultClassNames.caption\_label,

),

table: 'w-full border-collapse',

weekdays: cn('flex', defaultClassNames.weekdays),

weekday: cn(

'text-muted-foreground flex-1 select-none rounded-md text-[0.8rem] font-normal',

defaultClassNames.weekday,

),

week: cn('mt-2 flex w-full', defaultClassNames.week),

week\_number\_header: cn(

'w-[--cell-size] select-none',

defaultClassNames.week\_number\_header,

),

week\_number: cn(

'text-muted-foreground select-none text-[0.8rem]',

defaultClassNames.week\_number,

),

day: cn(

'group/day relative aspect-square h-full w-full select-none p-0 text-center [&:first-child[data-selected=true]\_button]:rounded-l-md [&:last-child[data-selected=true]\_button]:rounded-r-md',

defaultClassNames.day,

),

range\_start: cn(

'bg-accent rounded-l-md',

defaultClassNames.range\_start,

),

range\_middle: cn('rounded-none', defaultClassNames.range\_middle),

range\_end: cn('bg-accent rounded-r-md', defaultClassNames.range\_end),

today: cn(

'bg-accent text-accent-foreground rounded-md data-[selected=true]:rounded-none',

defaultClassNames.today,

),

outside: cn(

'text-muted-foreground aria-selected:text-muted-foreground',

defaultClassNames.outside,

),

disabled: cn(

'text-muted-foreground opacity-50',

defaultClassNames.disabled,

),

hidden: cn('invisible', defaultClassNames.hidden),

...classNames,

}}

components={{

Root: ({ className, rootRef, ...props }) => {

return (

<div

data-slot="calendar"

ref={rootRef}

className={cn(className)}

{...props}

/>

);

},

Chevron: ({ className, orientation, ...props }) => {

if (orientation === 'left') {

return (

<ChevronLeftIcon className={cn('size-4', className)} {...props} />

);

}

if (orientation === 'right') {

return (

<ChevronRightIcon

className={cn('size-4', className)}

{...props}

/>

);

}

return (

<ChevronDownIcon className={cn('size-4', className)} {...props} />

);

},

DayButton: CalendarDayButton,

WeekNumber: ({ children, ...props }) => {

return (

<td {...props}>

<div className="flex size-[--cell-size] items-center justify-center text-center">

{children}

</div>

</td>

);

},

...components,

}}

{...props}

/>

);

}

function CalendarDayButton({

className,

day,

modifiers,

...props

}: React.ComponentProps<typeof DayButton>) {

const defaultClassNames = getDefaultClassNames();

const ref = React.useRef<HTMLButtonElement>(null);

React.useEffect(() => {

if (modifiers.focused) ref.current?.focus();

}, [modifiers.focused]);

return (

<Button

ref={ref}

variant="ghost"

size="icon"

data-day={day.date.toLocaleDateString()}

data-selected-single={

modifiers.selected &&

!modifiers.range\_start &&

!modifiers.range\_end &&

!modifiers.range\_middle

}

data-range-start={modifiers.range\_start}

data-range-end={modifiers.range\_end}

data-range-middle={modifiers.range\_middle}

className={cn(

'data-[selected-single=true]:bg-primary data-[selected-single=true]:text-primary-foreground data-[range-middle=true]:bg-accent data-[range-middle=true]:text-accent-foreground data-[range-start=true]:bg-primary data-[range-start=true]:text-primary-foreground data-[range-end=true]:bg-primary data-[range-end=true]:text-primary-foreground group-data-[focused=true]/day:border-ring group-data-[focused=true]/day:ring-ring/50 flex aspect-square h-auto w-full min-w-[--cell-size] flex-col gap-1 font-normal leading-none data-[range-end=true]:rounded-md data-[range-middle=true]:rounded-none data-[range-start=true]:rounded-md group-data-[focused=true]/day:relative group-data-[focused=true]/day:z-10 group-data-[focused=true]/day:ring-[3px] [&>span]:text-xs [&>span]:opacity-70',

defaultClassNames.day,

className,

)}

{...props}

/>

);

}

export { Calendar, CalendarDayButton };

Card.tsx

import \* as React from 'react';

import { cn } from '@/lib/utils';

const Card = React.forwardRef<

HTMLDivElement,

React.HTMLAttributes<HTMLDivElement>

>(({ className, ...props }, ref) => (

<div

ref={ref}

className={cn(

'rounded-xl border bg-card text-card-foreground shadow',

className,

)}

{...props}

/>

));

Card.displayName = 'Card';

const CardHeader = React.forwardRef<

HTMLDivElement,

React.HTMLAttributes<HTMLDivElement>

>(({ className, ...props }, ref) => (

<div

ref={ref}

className={cn('flex flex-col space-y-1.5 p-6', className)}

{...props}

/>

));

CardHeader.displayName = 'CardHeader';

const CardTitle = React.forwardRef<

HTMLDivElement,

React.HTMLAttributes<HTMLDivElement>

>(({ className, ...props }, ref) => (

<div

ref={ref}

className={cn('font-semibold leading-none tracking-tight', className)}

{...props}

/>

));

CardTitle.displayName = 'CardTitle';

const CardDescription = React.forwardRef<

HTMLDivElement,

React.HTMLAttributes<HTMLDivElement>

>(({ className, ...props }, ref) => (

<div

ref={ref}

className={cn('text-sm text-muted-foreground', className)}

{...props}

/>

));

CardDescription.displayName = 'CardDescription';

const CardContent = React.forwardRef<

HTMLDivElement,

React.HTMLAttributes<HTMLDivElement>

>(({ className, ...props }, ref) => (

<div ref={ref} className={cn('p-6 pt-0', className)} {...props} />

));

CardContent.displayName = 'CardContent';

const CardFooter = React.forwardRef<

HTMLDivElement,

React.HTMLAttributes<HTMLDivElement>

>(({ className, ...props }, ref) => (

<div

ref={ref}

className={cn('flex items-center p-6 pt-0', className)}

{...props}

/>

));

CardFooter.displayName = 'CardFooter';

export {

Card,

CardHeader,

CardFooter,

CardTitle,

CardDescription,

CardContent,

};

checkbox.tsx

import \* as React from 'react';

import \* as CheckboxPrimitive from '@radix-ui/react-checkbox';

import { Check } from 'lucide-react';

import { cn } from '@/lib/utils';

const Checkbox = React.forwardRef<

React.ElementRef<typeof CheckboxPrimitive.Root>,

React.ComponentPropsWithoutRef<typeof CheckboxPrimitive.Root>

>(({ className, ...props }, ref) => (

<CheckboxPrimitive.Root

ref={ref}

className={cn(

'peer h-4 w-4 shrink-0 rounded-sm border border-primary shadow focus-visible:outline-none focus-visible:ring-1 focus-visible:ring-ring disabled:cursor-not-allowed disabled:opacity-50 data-[state=checked]:bg-primary data-[state=checked]:text-primary-foreground',

className,

)}

{...props}

>

<CheckboxPrimitive.Indicator

className={cn('flex items-center justify-center text-current')}

>

<Check className="h-4 w-4" />

</CheckboxPrimitive.Indicator>

</CheckboxPrimitive.Root>

));

Checkbox.displayName = CheckboxPrimitive.Root.displayName;

export { Checkbox };

Command.tsx

import \* as React from 'react';

import { type DialogProps } from '@radix-ui/react-dialog';

import { Command as CommandPrimitive } from 'cmdk';

import { Search } from 'lucide-react';

import { cn } from '@/lib/utils';

import { Dialog, DialogContent } from '@/components/ui/dialog';

const Command = React.forwardRef<

React.ElementRef<typeof CommandPrimitive>,

React.ComponentPropsWithoutRef<typeof CommandPrimitive>

>(({ className, ...props }, ref) => (

<CommandPrimitive

ref={ref}

className={cn(

'flex h-full w-full flex-col overflow-hidden rounded-md bg-popover text-popover-foreground',

className,

)}

{...props}

/>

));

Command.displayName = CommandPrimitive.displayName;

const CommandDialog = ({ children, ...props }: DialogProps) => {

return (

<Dialog {...props}>

<DialogContent className="overflow-hidden p-0">

<Command className="[&\_[cmdk-group-heading]]:px-2 [&\_[cmdk-group-heading]]:font-medium [&\_[cmdk-group-heading]]:text-muted-foreground [&\_[cmdk-group]:not([hidden])\_~[cmdk-group]]:pt-0 [&\_[cmdk-group]]:px-2 [&\_[cmdk-input-wrapper]\_svg]:h-5 [&\_[cmdk-input-wrapper]\_svg]:w-5 [&\_[cmdk-input]]:h-12 [&\_[cmdk-item]]:px-2 [&\_[cmdk-item]]:py-3 [&\_[cmdk-item]\_svg]:h-5 [&\_[cmdk-item]\_svg]:w-5">

{children}

</Command>

</DialogContent>

</Dialog>

);

};

const CommandInput = React.forwardRef<

React.ElementRef<typeof CommandPrimitive.Input>,

React.ComponentPropsWithoutRef<typeof CommandPrimitive.Input>

>(({ className, ...props }, ref) => (

<div className="flex items-center border-b px-3" cmdk-input-wrapper="">

<Search className="mr-2 h-4 w-4 shrink-0 opacity-50" />

<CommandPrimitive.Input

ref={ref}

className={cn(

'flex h-10 w-full rounded-md bg-transparent py-3 text-sm outline-none placeholder:text-muted-foreground disabled:cursor-not-allowed disabled:opacity-50',

className,

)}

{...props}

/>

</div>

));

CommandInput.displayName = CommandPrimitive.Input.displayName;

const CommandList = React.forwardRef<

React.ElementRef<typeof CommandPrimitive.List>,

React.ComponentPropsWithoutRef<typeof CommandPrimitive.List>

>(({ className, ...props }, ref) => (

<CommandPrimitive.List

ref={ref}

className={cn('max-h-[300px] overflow-y-auto overflow-x-hidden', className)}

{...props}

/>

));

CommandList.displayName = CommandPrimitive.List.displayName;

const CommandEmpty = React.forwardRef<

React.ElementRef<typeof CommandPrimitive.Empty>,

React.ComponentPropsWithoutRef<typeof CommandPrimitive.Empty>

>((props, ref) => (

<CommandPrimitive.Empty

ref={ref}

className="py-6 text-center text-sm"

{...props}

/>

));

CommandEmpty.displayName = CommandPrimitive.Empty.displayName;

const CommandGroup = React.forwardRef<

React.ElementRef<typeof CommandPrimitive.Group>,

React.ComponentPropsWithoutRef<typeof CommandPrimitive.Group>

>(({ className, ...props }, ref) => (

<CommandPrimitive.Group

ref={ref}

className={cn(

'overflow-hidden p-1 text-foreground [&\_[cmdk-group-heading]]:px-2 [&\_[cmdk-group-heading]]:py-1.5 [&\_[cmdk-group-heading]]:text-xs [&\_[cmdk-group-heading]]:font-medium [&\_[cmdk-group-heading]]:text-muted-foreground',

className,

)}

{...props}

/>

));

CommandGroup.displayName = CommandPrimitive.Group.displayName;

const CommandSeparator = React.forwardRef<

React.ElementRef<typeof CommandPrimitive.Separator>,

React.ComponentPropsWithoutRef<typeof CommandPrimitive.Separator>

>(({ className, ...props }, ref) => (

<CommandPrimitive.Separator

ref={ref}

className={cn('-mx-1 h-px bg-border', className)}

{...props}

/>

));

CommandSeparator.displayName = CommandPrimitive.Separator.displayName;

const CommandItem = React.forwardRef<

React.ElementRef<typeof CommandPrimitive.Item>,

React.ComponentPropsWithoutRef<typeof CommandPrimitive.Item>

>(({ className, ...props }, ref) => (

<CommandPrimitive.Item

ref={ref}

className={cn(

'relative flex cursor-default gap-2 select-none items-center rounded-sm px-2 py-1.5 text-sm outline-none data-[disabled=true]:pointer-events-none data-[selected=true]:bg-accent data-[selected=true]:text-accent-foreground data-[disabled=true]:opacity-50 [&\_svg]:pointer-events-none [&\_svg]:size-4 [&\_svg]:shrink-0',

className,

)}

{...props}

/>

));

CommandItem.displayName = CommandPrimitive.Item.displayName;

const CommandShortcut = ({

className,

...props

}: React.HTMLAttributes<HTMLSpanElement>) => {

return (

<span

className={cn(

'ml-auto text-xs tracking-widest text-muted-foreground',

className,

)}

{...props}

/>

);

};

CommandShortcut.displayName = 'CommandShortcut';

export {

Command,

CommandDialog,

CommandInput,

CommandList,

CommandEmpty,

CommandGroup,

CommandItem,

CommandShortcut,

CommandSeparator,

};

Dialog.tsx

'use client';

import \* as React from 'react';

import \* as DialogPrimitive from '@radix-ui/react-dialog';

import { X } from 'lucide-react';

import { cn } from '@/lib/utils';

const Dialog = DialogPrimitive.Root;

const DialogTrigger = DialogPrimitive.Trigger;

const DialogPortal = DialogPrimitive.Portal;

const DialogClose = DialogPrimitive.Close;

const DialogOverlay = React.forwardRef<

React.ElementRef<typeof DialogPrimitive.Overlay>,

React.ComponentPropsWithoutRef<typeof DialogPrimitive.Overlay>

>(({ className, ...props }, ref) => (

<DialogPrimitive.Overlay

ref={ref}

className={cn(

'fixed inset-0 z-50 bg-black/80 data-[state=open]:animate-in data-[state=closed]:animate-out data-[state=closed]:fade-out-0 data-[state=open]:fade-in-0',

className,

)}

{...props}

/>

));

DialogOverlay.displayName = DialogPrimitive.Overlay.displayName;

const DialogContent = React.forwardRef<

React.ElementRef<typeof DialogPrimitive.Content>,

React.ComponentPropsWithoutRef<typeof DialogPrimitive.Content>

>(({ className, children, ...props }, ref) => (

<DialogPortal>

<DialogOverlay />

<DialogPrimitive.Content

ref={ref}

className={cn(

'fixed left-[50%] top-[50%] z-50 grid w-full max-w-lg translate-x-[-50%] translate-y-[-50%] gap-4 border bg-background p-6 shadow-lg duration-200 data-[state=open]:animate-in data-[state=closed]:animate-out data-[state=closed]:fade-out-0 data-[state=open]:fade-in-0 data-[state=closed]:zoom-out-95 data-[state=open]:zoom-in-95 data-[state=closed]:slide-out-to-left-1/2 data-[state=closed]:slide-out-to-top-[48%] data-[state=open]:slide-in-from-left-1/2 data-[state=open]:slide-in-from-top-[48%] sm:rounded-lg',

className,

)}

{...props}

>

{children}

<DialogPrimitive.Close className="absolute right-4 top-4 rounded-sm opacity-70 ring-offset-background transition-opacity hover:opacity-100 focus:outline-none focus:ring-2 focus:ring-ring focus:ring-offset-2 disabled:pointer-events-none data-[state=open]:bg-accent data-[state=open]:text-muted-foreground">

<X className="h-4 w-4" />

<span className="sr-only">Close</span>

</DialogPrimitive.Close>

</DialogPrimitive.Content>

</DialogPortal>

));

DialogContent.displayName = DialogPrimitive.Content.displayName;

const DialogHeader = ({

className,

...props

}: React.HTMLAttributes<HTMLDivElement>) => (

<div

className={cn(

'flex flex-col space-y-1.5 text-center sm:text-left',

className,

)}

{...props}

/>

);

DialogHeader.displayName = 'DialogHeader';

const DialogFooter = ({

className,

...props

}: React.HTMLAttributes<HTMLDivElement>) => (

<div

className={cn(

'flex flex-col-reverse sm:flex-row sm:justify-end sm:space-x-2',

className,

)}

{...props}

/>

);

DialogFooter.displayName = 'DialogFooter';

const DialogTitle = React.forwardRef<

React.ElementRef<typeof DialogPrimitive.Title>,

React.ComponentPropsWithoutRef<typeof DialogPrimitive.Title>

>(({ className, ...props }, ref) => (

<DialogPrimitive.Title

ref={ref}

className={cn(

'text-lg font-semibold leading-none tracking-tight',

className,

)}

{...props}

/>

));

DialogTitle.displayName = DialogPrimitive.Title.displayName;

const DialogDescription = React.forwardRef<

React.ElementRef<typeof DialogPrimitive.Description>,

React.ComponentPropsWithoutRef<typeof DialogPrimitive.Description>

>(({ className, ...props }, ref) => (

<DialogPrimitive.Description

ref={ref}

className={cn('text-sm text-muted-foreground', className)}

{...props}

/>

));

DialogDescription.displayName = DialogPrimitive.Description.displayName;

export {

Dialog,

DialogPortal,

DialogOverlay,

DialogTrigger,

DialogClose,

DialogContent,

DialogHeader,

DialogFooter,

DialogTitle,

DialogDescription,

};

Input.tsx

import \* as React from 'react';

import { cn } from '@/lib/utils';

const Input = React.forwardRef<HTMLInputElement, React.ComponentProps<'input'>>(

({ className, type, ...props }, ref) => {

return (

<input

type={type}

className={cn(

'flex h-9 w-full rounded-md border border-input bg-transparent px-3 py-1 text-base shadow-sm transition-colors file:border-0 file:bg-transparent file:text-sm file:font-medium file:text-foreground placeholder:text-muted-foreground focus-visible:outline-none focus-visible:ring-1 focus-visible:ring-ring disabled:cursor-not-allowed disabled:opacity-50 md:text-sm',

className,

)}

ref={ref}

{...props}

/>

);

},

);

Input.displayName = 'Input';

export { Input };

Label.tsx

import \* as React from 'react';

import \* as LabelPrimitive from '@radix-ui/react-label';

import { cva, type VariantProps } from 'class-variance-authority';

import { cn } from '@/lib/utils';

const labelVariants = cva(

'text-sm font-medium leading-none peer-disabled:cursor-not-allowed peer-disabled:opacity-70',

);

const Label = React.forwardRef<

React.ElementRef<typeof LabelPrimitive.Root>,

React.ComponentPropsWithoutRef<typeof LabelPrimitive.Root> &

VariantProps<typeof labelVariants>

>(({ className, ...props }, ref) => (

<LabelPrimitive.Root

ref={ref}

className={cn(labelVariants(), className)}

{...props}

/>

));

Label.displayName = LabelPrimitive.Root.displayName;

export { Label };

Popover.tsx

import \* as React from 'react';

import \* as PopoverPrimitive from '@radix-ui/react-popover';

import { cn } from '@/lib/utils';

const Popover = PopoverPrimitive.Root;

const PopoverTrigger = PopoverPrimitive.Trigger;

const PopoverAnchor = PopoverPrimitive.Anchor;

const PopoverContent = React.forwardRef<

React.ElementRef<typeof PopoverPrimitive.Content>,

React.ComponentPropsWithoutRef<typeof PopoverPrimitive.Content>

>(({ className, align = 'center', sideOffset = 4, ...props }, ref) => (

<PopoverPrimitive.Portal>

<PopoverPrimitive.Content

ref={ref}

align={align}

sideOffset={sideOffset}

className={cn(

'z-50 w-72 rounded-md border bg-popover p-4 text-popover-foreground shadow-md outline-none data-[state=open]:animate-in data-[state=closed]:animate-out data-[state=closed]:fade-out-0 data-[state=open]:fade-in-0 data-[state=closed]:zoom-out-95 data-[state=open]:zoom-in-95 data-[side=bottom]:slide-in-from-top-2 data-[side=left]:slide-in-from-right-2 data-[side=right]:slide-in-from-left-2 data-[side=top]:slide-in-from-bottom-2',

className,

)}

{...props}

/>

</PopoverPrimitive.Portal>

));

PopoverContent.displayName = PopoverPrimitive.Content.displayName;

export { Popover, PopoverTrigger, PopoverContent, PopoverAnchor };

Progress.tsx

import \* as React from 'react';

import \* as ProgressPrimitive from '@radix-ui/react-progress';

import { cn } from '@/lib/utils';

const Progress = React.forwardRef<

React.ElementRef<typeof ProgressPrimitive.Root>,

React.ComponentPropsWithoutRef<typeof ProgressPrimitive.Root>

>(({ className, value, ...props }, ref) => (

<ProgressPrimitive.Root

ref={ref}

className={cn(

'relative h-2 w-full overflow-hidden rounded-full bg-primary/20',

className,

)}

{...props}

>

<ProgressPrimitive.Indicator

className="h-full w-full flex-1 bg-primary transition-all"

style={{ transform: `translateX(-${100 - (value || 0)}%)` }}

/>

</ProgressPrimitive.Root>

));

Progress.displayName = ProgressPrimitive.Root.displayName;

export { Progress };

Select.tsx

import \* as React from 'react';

import \* as SelectPrimitive from '@radix-ui/react-select';

import { Check, ChevronDown, ChevronUp } from 'lucide-react';

import { cn } from '@/lib/utils';

const Select = SelectPrimitive.Root;

const SelectGroup = SelectPrimitive.Group;

const SelectValue = SelectPrimitive.Value;

const SelectTrigger = React.forwardRef<

React.ElementRef<typeof SelectPrimitive.Trigger>,

React.ComponentPropsWithoutRef<typeof SelectPrimitive.Trigger>

>(({ className, children, ...props }, ref) => (

<SelectPrimitive.Trigger

ref={ref}

className={cn(

'flex h-9 w-full items-center justify-between whitespace-nowrap rounded-md border border-input bg-transparent px-3 py-2 text-sm shadow-sm ring-offset-background placeholder:text-muted-foreground focus:outline-none focus:ring-1 focus:ring-ring disabled:cursor-not-allowed disabled:opacity-50 [&>span]:line-clamp-1',

className,

)}

{...props}

>

{children}

<SelectPrimitive.Icon asChild>

<ChevronDown className="h-4 w-4 opacity-50" />

</SelectPrimitive.Icon>

</SelectPrimitive.Trigger>

));

SelectTrigger.displayName = SelectPrimitive.Trigger.displayName;

const SelectScrollUpButton = React.forwardRef<

React.ElementRef<typeof SelectPrimitive.ScrollUpButton>,

React.ComponentPropsWithoutRef<typeof SelectPrimitive.ScrollUpButton>

>(({ className, ...props }, ref) => (

<SelectPrimitive.ScrollUpButton

ref={ref}

className={cn(

'flex cursor-default items-center justify-center py-1',

className,

)}

{...props}

>

<ChevronUp className="h-4 w-4" />

</SelectPrimitive.ScrollUpButton>

));

SelectScrollUpButton.displayName = SelectPrimitive.ScrollUpButton.displayName;

const SelectScrollDownButton = React.forwardRef<

React.ElementRef<typeof SelectPrimitive.ScrollDownButton>,

React.ComponentPropsWithoutRef<typeof SelectPrimitive.ScrollDownButton>

>(({ className, ...props }, ref) => (

<SelectPrimitive.ScrollDownButton

ref={ref}

className={cn(

'flex cursor-default items-center justify-center py-1',

className,

)}

{...props}

>

<ChevronDown className="h-4 w-4" />

</SelectPrimitive.ScrollDownButton>

));

SelectScrollDownButton.displayName =

SelectPrimitive.ScrollDownButton.displayName;

const SelectContent = React.forwardRef<

React.ElementRef<typeof SelectPrimitive.Content>,

React.ComponentPropsWithoutRef<typeof SelectPrimitive.Content>

>(({ className, children, position = 'popper', ...props }, ref) => (

<SelectPrimitive.Portal>

<SelectPrimitive.Content

ref={ref}

className={cn(

'relative z-50 max-h-96 min-w-[8rem] overflow-hidden rounded-md border bg-popover text-popover-foreground shadow-md data-[state=open]:animate-in data-[state=closed]:animate-out data-[state=closed]:fade-out-0 data-[state=open]:fade-in-0 data-[state=closed]:zoom-out-95 data-[state=open]:zoom-in-95 data-[side=bottom]:slide-in-from-top-2 data-[side=left]:slide-in-from-right-2 data-[side=right]:slide-in-from-left-2 data-[side=top]:slide-in-from-bottom-2',

position === 'popper' &&

'data-[side=bottom]:translate-y-1 data-[side=left]:-translate-x-1 data-[side=right]:translate-x-1 data-[side=top]:-translate-y-1',

className,

)}

position={position}

{...props}

>

<SelectScrollUpButton />

<SelectPrimitive.Viewport

className={cn(

'p-1',

position === 'popper' &&

'h-[var(--radix-select-trigger-height)] w-full min-w-[var(--radix-select-trigger-width)]',

)}

>

{children}

</SelectPrimitive.Viewport>

<SelectScrollDownButton />

</SelectPrimitive.Content>

</SelectPrimitive.Portal>

));

SelectContent.displayName = SelectPrimitive.Content.displayName;

const SelectLabel = React.forwardRef<

React.ElementRef<typeof SelectPrimitive.Label>,

React.ComponentPropsWithoutRef<typeof SelectPrimitive.Label>

>(({ className, ...props }, ref) => (

<SelectPrimitive.Label

ref={ref}

className={cn('px-2 py-1.5 text-sm font-semibold', className)}

{...props}

/>

));

SelectLabel.displayName = SelectPrimitive.Label.displayName;

const SelectItem = React.forwardRef<

React.ElementRef<typeof SelectPrimitive.Item>,

React.ComponentPropsWithoutRef<typeof SelectPrimitive.Item>

>(({ className, children, ...props }, ref) => (

<SelectPrimitive.Item

ref={ref}

className={cn(

'relative flex w-full cursor-default select-none items-center rounded-sm py-1.5 pl-2 pr-8 text-sm outline-none focus:bg-accent focus:text-accent-foreground data-[disabled]:pointer-events-none data-[disabled]:opacity-50',

className,

)}

{...props}

>

<span className="absolute right-2 flex h-3.5 w-3.5 items-center justify-center">

<SelectPrimitive.ItemIndicator>

<Check className="h-4 w-4" />

</SelectPrimitive.ItemIndicator>

</span>

<SelectPrimitive.ItemText>{children}</SelectPrimitive.ItemText>

</SelectPrimitive.Item>

));

SelectItem.displayName = SelectPrimitive.Item.displayName;

const SelectSeparator = React.forwardRef<

React.ElementRef<typeof SelectPrimitive.Separator>,

React.ComponentPropsWithoutRef<typeof SelectPrimitive.Separator>

>(({ className, ...props }, ref) => (

<SelectPrimitive.Separator

ref={ref}

className={cn('-mx-1 my-1 h-px bg-muted', className)}

{...props}

/>

));

SelectSeparator.displayName = SelectPrimitive.Separator.displayName;

export {

Select,

SelectGroup,

SelectValue,

SelectTrigger,

SelectContent,

SelectLabel,

SelectItem,

SelectSeparator,

SelectScrollUpButton,

SelectScrollDownButton,

};

Slider.tsx

import \* as React from 'react';

import \* as SliderPrimitive from '@radix-ui/react-slider';

import { cn } from '@/lib/utils';

const Slider = React.forwardRef<

React.ElementRef<typeof SliderPrimitive.Root>,

React.ComponentPropsWithoutRef<typeof SliderPrimitive.Root>

>(({ className, ...props }, ref) => (

<SliderPrimitive.Root

ref={ref}

className={cn(

'relative flex w-full touch-none select-none items-center',

className,

)}

{...props}

>

<SliderPrimitive.Track className="relative h-1.5 w-full grow overflow-hidden rounded-full bg-primary/20">

<SliderPrimitive.Range className="absolute h-full bg-primary" />

</SliderPrimitive.Track>

<SliderPrimitive.Thumb className="block h-4 w-4 rounded-full border border-primary/50 bg-background shadow transition-colors focus-visible:outline-none focus-visible:ring-1 focus-visible:ring-ring disabled:pointer-events-none disabled:opacity-50" />

</SliderPrimitive.Root>

));

Slider.displayName = SliderPrimitive.Root.displayName;

export { Slider };

Switch.tsx

import \* as React from 'react';

import \* as SwitchPrimitives from '@radix-ui/react-switch';

import { cn } from '@/lib/utils';

const Switch = React.forwardRef<

React.ElementRef<typeof SwitchPrimitives.Root>,

React.ComponentPropsWithoutRef<typeof SwitchPrimitives.Root>

>(({ className, ...props }, ref) => (

<SwitchPrimitives.Root

className={cn(

'peer inline-flex h-5 w-9 shrink-0 cursor-pointer items-center rounded-full border-2 border-transparent shadow-sm transition-colors focus-visible:outline-none focus-visible:ring-2 focus-visible:ring-ring focus-visible:ring-offset-2 focus-visible:ring-offset-background disabled:cursor-not-allowed disabled:opacity-50 data-[state=checked]:bg-primary data-[state=unchecked]:bg-input',

className,

)}

{...props}

ref={ref}

>

<SwitchPrimitives.Thumb

className={cn(

'pointer-events-none block h-4 w-4 rounded-full bg-background shadow-lg ring-0 transition-transform data-[state=checked]:translate-x-4 data-[state=unchecked]:translate-x-0',

)}

/>

</SwitchPrimitives.Root>

));

Switch.displayName = SwitchPrimitives.Root.displayName;

export { Switch };

Table.tsx

import \* as React from 'react';

import { cn } from '@/lib/utils';

const Table = React.forwardRef<

HTMLTableElement,

React.HTMLAttributes<HTMLTableElement>

>(({ className, ...props }, ref) => (

<div className="relative w-full overflow-auto">

<table

ref={ref}

className={cn('w-full caption-bottom text-sm', className)}

{...props}

/>

</div>

));

Table.displayName = 'Table';

const TableHeader = React.forwardRef<

HTMLTableSectionElement,

React.HTMLAttributes<HTMLTableSectionElement>

>(({ className, ...props }, ref) => (

<thead ref={ref} className={cn('[&\_tr]:border-b', className)} {...props} />

));

TableHeader.displayName = 'TableHeader';

const TableBody = React.forwardRef<

HTMLTableSectionElement,

React.HTMLAttributes<HTMLTableSectionElement>

>(({ className, ...props }, ref) => (

<tbody

ref={ref}

className={cn('[&\_tr:last-child]:border-0', className)}

{...props}

/>

));

TableBody.displayName = 'TableBody';

const TableFooter = React.forwardRef<

HTMLTableSectionElement,

React.HTMLAttributes<HTMLTableSectionElement>

>(({ className, ...props }, ref) => (

<tfoot

ref={ref}

className={cn(

'border-t bg-muted/50 font-medium [&>tr]:last:border-b-0',

className,

)}

{...props}

/>

));

TableFooter.displayName = 'TableFooter';

const TableRow = React.forwardRef<

HTMLTableRowElement,

React.HTMLAttributes<HTMLTableRowElement>

>(({ className, ...props }, ref) => (

<tr

ref={ref}

className={cn(

'border-b transition-colors hover:bg-muted/50 data-[state=selected]:bg-muted',

className,

)}

{...props}

/>

));

TableRow.displayName = 'TableRow';

const TableHead = React.forwardRef<

HTMLTableCellElement,

React.ThHTMLAttributes<HTMLTableCellElement>

>(({ className, ...props }, ref) => (

<th

ref={ref}

className={cn(

'h-10 px-2 text-left align-middle font-medium text-muted-foreground [&:has([role=checkbox])]:pr-0 [&>[role=checkbox]]:translate-y-[2px]',

className,

)}

{...props}

/>

));

TableHead.displayName = 'TableHead';

const TableCell = React.forwardRef<

HTMLTableCellElement,

React.TdHTMLAttributes<HTMLTableCellElement>

>(({ className, ...props }, ref) => (

<td

ref={ref}

className={cn(

'p-2 align-middle [&:has([role=checkbox])]:pr-0 [&>[role=checkbox]]:translate-y-[2px]',

className,

)}

{...props}

/>

));

TableCell.displayName = 'TableCell';

const TableCaption = React.forwardRef<

HTMLTableCaptionElement,

React.HTMLAttributes<HTMLTableCaptionElement>

>(({ className, ...props }, ref) => (

<caption

ref={ref}

className={cn('mt-4 text-sm text-muted-foreground', className)}

{...props}

/>

));

TableCaption.displayName = 'TableCaption';

export {

Table,

TableHeader,

TableBody,

TableFooter,

TableHead,

TableRow,

TableCell,

TableCaption,

};

Toggle.tsx

import \* as React from 'react';

import \* as TogglePrimitive from '@radix-ui/react-toggle';

import { cva, type VariantProps } from 'class-variance-authority';

import { cn } from '@/lib/utils';

const toggleVariants = cva(

'inline-flex items-center justify-center gap-2 rounded-md text-sm font-medium transition-colors hover:bg-muted hover:text-muted-foreground focus-visible:outline-none focus-visible:ring-1 focus-visible:ring-ring disabled:pointer-events-none disabled:opacity-50 data-[state=on]:bg-accent data-[state=on]:text-accent-foreground [&\_svg]:pointer-events-none [&\_svg]:size-4 [&\_svg]:shrink-0',

{

variants: {

variant: {

default: 'bg-transparent',

outline:

'border border-input bg-transparent shadow-sm hover:bg-accent hover:text-accent-foreground',

},

size: {

default: 'h-9 px-2 min-w-9',

sm: 'h-8 px-1.5 min-w-8',

lg: 'h-10 px-2.5 min-w-10',

},

},

defaultVariants: {

variant: 'default',

size: 'default',

},

},

);

const Toggle = React.forwardRef<

React.ElementRef<typeof TogglePrimitive.Root>,

React.ComponentPropsWithoutRef<typeof TogglePrimitive.Root> &

VariantProps<typeof toggleVariants>

>(({ className, variant, size, ...props }, ref) => (

<TogglePrimitive.Root

ref={ref}

className={cn(toggleVariants({ variant, size, className }))}

{...props}

/>

));

Toggle.displayName = TogglePrimitive.Root.displayName;

export { Toggle, toggleVariants };

Tooltip.tsx

import \* as React from 'react';

import \* as TooltipPrimitive from '@radix-ui/react-tooltip';

import { cn } from '@/lib/utils';

const TooltipProvider = TooltipPrimitive.Provider;

const Tooltip = TooltipPrimitive.Root;

const TooltipTrigger = TooltipPrimitive.Trigger;

const TooltipContent = React.forwardRef<

React.ElementRef<typeof TooltipPrimitive.Content>,

React.ComponentPropsWithoutRef<typeof TooltipPrimitive.Content>

>(({ className, sideOffset = 4, ...props }, ref) => (

<TooltipPrimitive.Portal>

<TooltipPrimitive.Content

ref={ref}

sideOffset={sideOffset}

className={cn(

'z-50 overflow-hidden rounded-md bg-primary px-3 py-1.5 text-xs text-primary-foreground animate-in fade-in-0 zoom-in-95 data-[state=closed]:animate-out data-[state=closed]:fade-out-0 data-[state=closed]:zoom-out-95 data-[side=bottom]:slide-in-from-top-2 data-[side=left]:slide-in-from-right-2 data-[side=right]:slide-in-from-left-2 data-[side=top]:slide-in-from-bottom-2',

className,

)}

{...props}

/>

</TooltipPrimitive.Portal>

));

TooltipContent.displayName = TooltipPrimitive.Content.displayName;

export { Tooltip, TooltipTrigger, TooltipContent, TooltipProvider };

Layout.tsx

import React, { useState, useEffect } from 'react';

import { NavLink, useNavigate } from 'react-router-dom';

import StarryBackground from './ShootingStars';

const Layout = ({ children }: { children: React.ReactNode }) => {

const navigate = useNavigate();

const [satsBalance, setSatsBalance] = useState(0);

useEffect(() => {

const savedSats = localStorage.getItem('dp\_default\_sats\_balance');

setSatsBalance(Number(savedSats) || 0);

const handleStorageChange = () => {

const newSats = localStorage.getItem('dp\_default\_sats\_balance');

setSatsBalance(Number(newSats) || 0);

};

window.addEventListener('storage', handleStorageChange);

return () => window.removeEventListener('storage', handleStorageChange);

}, []);

const handleLogout = () => {

localStorage.removeItem('isLoggedIn');

localStorage.removeItem('owner');

localStorage.removeItem('email');

navigate('/login');

};

return (

<div className="driftpay-theme min-h-screen">

<StarryBackground />

<div className="wrap pt-12">

<nav className="row" style={{ justifyContent: 'center', marginBottom: '10px', gap: '0.5rem', background: 'rgba(10, 10, 35, 0.5)', backdropFilter: 'blur(4px)', padding: '8px', borderRadius: '16px' }}>

<NavLink to="/" className={({ isActive }) => `btn ghost ${isActive ? 'primary' : ''}`}>Home</NavLink>

<NavLink to="/journal" className={({ isActive }) => `btn ghost ${isActive ? 'primary' : ''}`}>Journal</NavLink>

<NavLink to="/stats" className={({ isActive }) => `btn ghost ${isActive ? 'primary' : ''}`}>Rewards</NavLink>

<NavLink to="/market" className={({ isActive }) => `btn ghost ${isActive ? 'primary' : ''}`}>Market</NavLink>

</nav>

<div className="flex justify-center mb-4">

<div className="bg-black/30 backdrop-blur-sm px-3 py-1 rounded-full text-sm">

⚡ {satsBalance.toLocaleString()} SATs

</div>

</div>

<main>{children}</main>

</div>

<div className="fixed bottom-4 left-4 z-20">

<button onClick={handleLogout} className="text-xs text-muted-foreground underline">

Logout

</button>

</div>

</div>

);

};

export default Layout;

ShootingStars.tsx

import React, { useEffect, useState } from 'react';

const StarryBackground = () => {

const [stars, setStars] = useState<React.CSSProperties[]>([]);

const [shootingStars, setShootingStars] = useState<React.CSSProperties[]>([]);

useEffect(() => {

// Drifting stars

const generateStars = () => {

const newStars = Array.from({ length: 50 }).map(() => {

const size = Math.random() \* 2 + 1; // 1px to 3px

const duration = Math.random() \* 20 + 15; // 15s to 35s

const rotation = (Math.random() - 0.5) \* 90; // -45 to 45 degrees

return {

left: `${Math.random() \* 100}%`,

width: `${size}px`,

height: `${size}px`,

animationName: 'drift-up',

animationDuration: `${duration}s`,

animationDelay: `${Math.random() \* duration}s`,

'--rotation': `${rotation}deg`,

} as React.CSSProperties;

});

setStars(newStars);

};

// Shooting stars

const generateShootingStars = () => {

const newShootingStars = Array.from({ length: 10 }).map(() => {

const animDuration = Math.random() \* 2 + 3; // 3s to 5s

return {

top: `${Math.random() \* 100}%`,

left: `${Math.random() \* 100}%`,

'--angle': `${Math.random() \* 360}deg`,

animationDuration: `${animDuration}s, ${animDuration}s`,

animationDelay: `${Math.random() \* 10}s`,

} as React.CSSProperties;

});

setShootingStars(newShootingStars);

};

generateStars();

generateShootingStars();

}, []);

return (

<div className="starry-background">

{stars.map((style, index) => (

<div key={`star-${index}`} className="star" style={style}></div>

))}

{shootingStars.map((style, index) => (

<div key={`shooting-star-${index}`} className="shooting-star" style={style}></div>

))}

</div>

);

};

export default StarryBackground;

* /data/
* /database.sqlite
* import React, { useEffect, useState } from 'react';
* const StarryBackground = () => {
* const [stars, setStars] = useState<React.CSSProperties[]>([]);
* const [shootingStars, setShootingStars] = useState<React.CSSProperties[]>([]);
* useEffect(() => {
* // Drifting stars
* const generateStars = () => {
* const newStars = Array.from({ length: 50 }).map(() => {
* const size = Math.random() \* 2 + 1; // 1px to 3px
* const duration = Math.random() \* 20 + 15; // 15s to 35s
* const rotation = (Math.random() - 0.5) \* 90; // -45 to 45 degrees
* return {
* left: `${Math.random() \* 100}%`,
* width: `${size}px`,
* height: `${size}px`,
* animationName: 'drift-up',
* animationDuration: `${duration}s`,
* animationDelay: `${Math.random() \* duration}s`,
* '--rotation': `${rotation}deg`,
* } as React.CSSProperties;
* });
* setStars(newStars);
* };
* // Shooting stars
* const generateShootingStars = () => {
* const newShootingStars = Array.from({ length: 10 }).map(() => {
* const animDuration = Math.random() \* 2 + 3; // 3s to 5s
* return {
* top: `${Math.random() \* 100}%`,
* left: `${Math.random() \* 100}%`,
* '--angle': `${Math.random() \* 360}deg`,
* animationDuration: `${animDuration}s, ${animDuration}s`,
* animationDelay: `${Math.random() \* 10}s`,
* } as React.CSSProperties;
* });
* setShootingStars(newShootingStars);
* };
* generateStars();
* generateShootingStars();
* }, []);
* return (
* <div className="starry-background">
* {stars.map((style, index) => (
* <div key={`star-${index}`} className="star" style={style}></div>
* ))}
* {shootingStars.map((style, index) => (
* <div key={`shooting-star-${index}`} className="shooting-star" style={style}></div>
* ))}
* </div>
* );
* };
* export default StarryBackground;
* data/

database.sqlite

U1FMaXRlIGZvcm1hdCAzABAAAQEAQCAgAAAAFwAAAAoAAAAAAAAAAAAAAAQAAAAEAAAAAAAAAAAAAAABAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAXAC6KEg0OkgAJCrkADuwOmg1rDmcOPAwUDTYKuQvbAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAgh8IBxclJQGEAXRhYmxldG9rZW5fbGVkZ2VydG9rZW5fbGVkZ2VyCUNSRUFURSBUQUJMRSB0b2tlbl9sZWRnZXIgKAogIGVudHJ5X2lkIFRFWFQgUFJJTUFSWSBLRVksCiAgdXNlcl9pZCBURVhUIE5PVCBOVUxMLAogIHRpbWVzdGFtcCBURVhUIE5PVCBOVUxMIERFRkFVTFQgQ1VSUkVOVF9USU1FU1RBTVAsCiAgc291cmNlIFRFWFQgTk9UIE5VTEwsCiAgYW1vdW50IElOVEVHRVIgTk9UIE5VTEwsCiAgbm90ZSBURVhULAogIEZPUkVJR04gS0VZICh1c2VyX2lkKSBSRUZFUkVOQ0VTIHVzZXJzKHVzZXJfaWQpCik3CQYXSyUBAGluZGV4c3FsaXRlX2F1dG9pbmRleF90b2tlbl9sZWRnZXJfMXRva2VuX2xlZGdlcgqCHwYHFyEhAYQJdGFibGVzbGVlcF9sb2dzc2xlZXBfbG9ncwdDUkVBVEUgVEFCTEUgc2xlZXBfbG9ncyAoCiAgbG9nX2lkIFRFWFQgUFJJTUFSWSBLRVksCiAgdXNlcl9pZCBURVhUIE5PVCBOVUxMLAogIHN0YXJ0X3RpbWUgVEVYVCBOT1QgTlVMTCwKICBlbmRfdGltZSBURVhUIE5PVCBOVUxMLAogIGR1cmF0aW9uIFJFQUwgTk9UIE5VTEwsCiAgdG9rZW5fcmV3YXJkZWQgQk9PTEVBTiBOT1QgTlVMTCBERUZBVUxUIDAsCiAgRk9SRUlHTiBLRVkgKHVzZXJfaWQpIFJFRkVSRU5DRVMgdXNlcnModXNlcl9pZCkKKTMHBhdHIQEAaW5kZXhzcWxpdGVfYXV0b2luZGV4X3NsZWVwX2xvZ3NfMXNsZWVwX2xvZ3MIgU4DBxcXFwGCe3RhYmxldXNlcnN1c2VycwRDUkVBVEUgVEFCTEUgdXNlcnMgKAogIHVzZXJfaWQgVEVYVCBQUklNQVJZIEtFWSwKICBlbWFpbCBURVhUIE5PVCBOVUxMIFVOSVFVRSwKICBwYXNzd29yZF9oYXNoIFRFWFQgTk9UIE5VTEwsCiAgY3JlYXRlZF9hdCBURVhUIE5PVCBOVUxMIERFRkFVTFQgQ1VSUkVOVF9USU1FU1RBTVAsCiAgbGFzdF9sb2dpbiBURVhUCikpBQYXPRcBAGluZGV4c3FsaXRlX2F1dG9pbmRleF91c2Vyc18ydXNlcnMGKQQGFz0XAQBpbmRleHNxbGl0ZV9hdXRvaW5kZXhfdXNlcnNfMXVzZXJzBQAAAAgAAAAAUAIGFysrAVl0YWJsZXNxbGl0ZV9zZXF1ZW5jZXNxbGl0ZV9zZXF1ZW5jZQNDUkVBVEUgVEFCTEUgc3FsaXRlX3NlcXVlbmNlKG5hbWUsc2VxKYIRAQcXKysBg1l0YWJsZWpvdXJuYWxfZW50cmllc2pvdXJuYWxfZW50cmllcwJDUkVBVEUgVEFCTEUgam91cm5hbF9lbnRyaWVzICgKICBpZCBJTlRFR0VSIFBSSU1BUlkgS0VZIEFVVE9JTkNSRU1FTlQsCiAgdXNlcl9pZCBURVhUIE5PVCBOVUxMLAogIHRleHQgVEVYVCBOT1QgTlVMTCwKICB0aGVtZSBURVhULAogIHBvZXRpY19saW5lIFRFWFQsCiAgdG9rZW5zX2Vhcm5lZCBJTlRFR0VSLAogIGNsYXJpdHlfZWFybmVkIElOVEVHRVIsCiAgdGltZXN0YW1wIFRFWFQgTk9UIE5VTEwKKQ0AAAABD7MAD7MPYw8JAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAE4GCQAbGwBLCAA9ZGVmYXVsdCBzYWRmYXNZb3VyIGRyZWFtcyBoYXZlIGJlZW4gcmVjb3JkZWQuMjAyNS0wOS0xN1QwNToxNDozMC4wOTdaAAAApxsbAEsIAEwICQAbFwBLCAA9ZGVmYXVsdEZyZXNoWW91ciBkcmVhbXMgaGF2ZSBiZWVuIHJlY29yZGVkLjIwMjUtMDktMTdUMDU6MjY6MjkuNjc2Wk0JSwoJABsVAEsIAD1kZWZhdWx0dGVzdFlvdXIgZHJlYW1zIGhhdmUgYmVlbiByZWNvcmRlZC4yMDI1LTA5LTE3VDIwOjU3OjQ5LjA3MVoNAAAAAQ/rAA/rAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAEwEDKwFqb3VybmFsX2VudHJpZXMKDQAAAAAQAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAoAAAAAEAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAKAAAAABAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAADQAAAAAQAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAoAAAAAEAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAANAAAAABAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAACgAAAAAQAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA==

* scripts/
* /[dev.ts](http://dev.ts)
* import { startServer } from '../server/index.js';
* import { createServer } from 'vite';
* let viteServer;
* async function startDev() {
* // Start the Express API server first
* await startServer(3001);
* // Then start Vite in dev mode
* const viteServer = await createServer({
* configFile: './vite.config.js',
* });
* const x = await viteServer.listen();
* console.log(
* `Vite dev server running on port ${viteServer.config.server.port}`,
* );
* }
* // Handle nodemon restarts - only needed if we're running under nodemon
* if (
* process.env.npm\_lifecycle\_event &&
* process.env.npm\_lifecycle\_event.includes('watch')
* ) {
* let isRestarting = false;
* process.once('SIGUSR2', async () => {
* if (isRestarting) return;
* isRestarting = true;
* console.log('Nodemon restart detected, closing Vite server...');
* if (viteServer) {
* try {
* await viteServer.close();
* console.log('Vite server closed successfully');
* } catch (err) {
* console.error('Error closing Vite server:', err);
* }
* }
* // Allow nodemon to restart the process
* process.kill(process.pid, 'SIGUSR2');
* });
* }
* startDev();
* server/
* db/
* [database.ts/](http://database.ts/)
* import { Kysely, SqliteDialect } from 'kysely';
* import Database from 'better-sqlite3';
* import { DB } from './types';
* const dialect = new SqliteDialect({
* database: new Database(process.env.DATA\_DIRECTORY ? `${process.env.DATA\_DIRECTORY}/database.sqlite` : 'data/database.sqlite'),
* });
* export const db = new Kysely<DB>({
* dialect,
* log: ['query', 'error'],
* });
* [types.ts/](http://types.ts/)
* import { Generated } from 'kysely';
* export interface JournalEntriesTable {
* id: Generated<number>;
* user\_id: string;
* text: string;
* theme: string | null;
* poetic\_line: string | null;
* tokens\_earned: number | null;
* clarity\_earned: number | null; // This can be removed in a future migration if no longer needed
* timestamp: string;
* }
* export interface UsersTable {
* user\_id: string;
* email: string;
* password\_hash: string;
* created\_at: string;
* last\_login: string | null;
* }
* export interface SleepLogsTable {
* log\_id: string;
* user\_id: string;
* start\_time: string;
* end\_time: string;
* duration: number;
* token\_rewarded: boolean;
* }
* export interface TokenLedgerTable {
* entry\_id: string;
* user\_id: string;
* timestamp: string;
* source: string;
* amount: number;
* note: string | null;
* }
* export interface DB {
* journal\_entries: JournalEntriesTable;
* users: UsersTable;
* sleep\_logs: SleepLogsTable;
* token\_ledger: TokenLedgerTable;
* }
* routes/
* [index.ts](http://index.ts)
* import express from 'express';
* import dotenv from 'dotenv';
* import { setupStaticServing } from './static-serve.js';
* import journalRouter from './routes/journal';
* import rewardsRouter from './routes/rewards';
* dotenv.config();
* const app = express();
* // Body parsing middleware
* app.use(express.json());
* app.use(express.urlencoded({ extended: true }));
* // API routes
* app.use('/api/journal', journalRouter);
* app.use('/api/rewards', rewardsRouter);
* // Export a function to start the server
* export async function startServer(port) {
* try {
* if (process.env.NODE\_ENV === 'production') {
* setupStaticServing(app);
* }
* app.listen(port, () => {
* console.log(`API Server running on port ${port}`);
* });
* } catch (err) {
* console.error('Failed to start server:', err);
* process.exit(1);
* }
* }
* // Start the server directly if this is the main module
* if (import.meta.url === `file://${process.argv[1]}`) {
* console.log('Starting server...');
* startServer(process.env.PORT || 3001);
* }
* [static-serve.ts](http://static-serve.ts)
* import path from 'path';
* import express from 'express';
* /\*\*
* \* Sets up static file serving for the Express app
* \* @param app Express application instance
* \*/
* export function setupStaticServing(app: express.Application) {
* // Serve static files from the public directory
* app.use(express.static(path.join(process.cwd(), 'public')));
* // For any other routes, serve the index.html file
* app.get('/{\*splat}', (req, res, next) => {
* // Skip API routes
* if (req.path.startsWith('/api/')) {
* return next();
* }
* res.sendFile(path.join(process.cwd(), 'public', 'index.html'));
* });
* }
* [journal.ts/](http://journal.ts/)
* import path from 'path';
* import express from 'express';
* /\*\*
* \* Sets up static file serving for the Express app
* \* @param app Express application instance
* \*/
* export function setupStaticServing(app: express.Application) {
* // Serve static files from the public directory
* app.use(express.static(path.join(process.cwd(), 'public')));
* // For any other routes, serve the index.html file
* app.get('/{\*splat}', (req, res, next) => {
* // Skip API routes
* if (req.path.startsWith('/api/')) {
* return next();
* }
* res.sendFile(path.join(process.cwd(), 'public', 'index.html'));
* });
* }
* [rewards.ts/](http://rewards.ts/)
* import express from 'express';
* const router = express.Router();
* // This endpoint is currently disabled as per user request.
* // It can be re-enabled when lightning address integrations are needed.
* router.post('/claim', async (req, res) => {
* res.status(503).json({ message: 'Lightning address integration is temporarily disabled.' });
* return;
* /\*
* const { lightningAddress, amount } = req.body;
* const apiKey = process.env.ZBD\_API\_KEY;
* if (!apiKey) {
* console.error('ZBD\_API\_KEY is not set.');
* res.status(500).json({ message: 'Server configuration error.' });
* return;
* }
* if (!lightningAddress || !amount) {
* res.status(400).json({ message: 'Lightning address and amount are required.' });
* return;
* }
* if (amount <= 0) {
* res.status(400).json({ message: 'Amount must be positive.' });
* return;
* }
* try {
* const zbdResponse = await fetch('https://api.zebedee.io/v0/ln-address/send-payment', {
* method: 'POST',
* headers: {
* 'Content-Type': 'application/json',
* 'apikey': apiKey,
* },
* body: JSON.stringify({
* lnAddress: lightningAddress,
* amount: (amount \* 1000).toString(), // ZBD API expects amount in millisats
* comment: 'Driftpay Rewards',
* }),
* });
* const data = await zbdResponse.json();
* if (!zbdResponse.ok || !data.success) {
* console.error('ZBD API Error:', data);
* throw new Error(data.message || 'Payment failed at ZBD.');
* }
* res.status(200).json({
* message: 'Payout successful!',
* amount: amount,
* transactionId: data.data.transactionId,
* });
* } catch (error) {
* console.error('Failed to process claim:', error);
* const errorMessage = error instanceof Error ? error.message : 'An unknown error occurred';
* res.status(500).json({ message: `Failed to process claim: ${errorMessage}` });
* }
* \*/
* });
* export default router;
* [index.ts/](http://index.ts/)
* import express from 'express';
* import dotenv from 'dotenv';
* import { setupStaticServing } from './static-serve.js';
* import journalRouter from './routes/journal';
* import rewardsRouter from './routes/rewards';
* dotenv.config();
* const app = express();
* // Body parsing middleware
* app.use(express.json());
* app.use(express.urlencoded({ extended: true }));
* // API routes
* app.use('/api/journal', journalRouter);
* app.use('/api/rewards', rewardsRouter);
* // Export a function to start the server
* export async function startServer(port) {
* try {
* if (process.env.NODE\_ENV === 'production') {
* setupStaticServing(app);
* }
* app.listen(port, () => {
* console.log(`API Server running on port ${port}`);
* });
* } catch (err) {
* console.error('Failed to start server:', err);
* process.exit(1);
* }
* }
* // Start the server directly if this is the main module
* if (import.meta.url === `file://${process.argv[1]}`) {
* console.log('Starting server...');
* startServer(process.env.PORT || 3001);
* }
* [static-serve.ts/](http://static-serve.ts/)
* import path from 'path';
* import express from 'express';
* /\*\*
* \* Sets up static file serving for the Express app
* \* @param app Express application instance
* \*/
* export function setupStaticServing(app: express.Application) {
* // Serve static files from the public directory
* app.use(express.static(path.join(process.cwd(), 'public')));
* // For any other routes, serve the index.html file
* app.get('/{\*splat}', (req, res, next) => {
* // Skip API routes
* if (req.path.startsWith('/api/')) {
* return next();
* }
* res.sendFile(path.join(process.cwd(), 'public', 'index.html'));
* });
* }
* components.json/
* {
* "$schema": "https://ui.shadcn.com/schema.json",
* "style": "new-york",
* "rsc": false,
* "tsx": true,
* "tailwind": {
* "config": "tailwind.config.js",
* "css": "client/src/index.css",
* "baseColor": "zinc",
* "cssVariables": true,
* "prefix": ""
* },
* "aliases": {
* "components": "@/components",
* "utils": "@/lib/utils",
* "ui": "@/components/ui",
* "lib": "@/lib",
* "hooks": "@/hooks"
* },
* "iconLibrary": "lucide"
* }
* [postcss.config.js](http://postcss.config.js)
* export default {
* plugins: {
* tailwindcss: {},
* autoprefixer: {},
* },
* };
* [tailwind.config.js](http://tailwind.config.js)
* export default {
* darkMode: ['class'],
* content: ['./client/index.html', './client/src/\*\*/\*.{js,jsx,ts,tsx}'],
* theme: {
* extend: {
* borderRadius: {
* lg: 'var(--radius)',
* md: 'calc(var(--radius) - 2px)',
* sm: 'calc(var(--radius) - 4px)',
* },
* colors: {
* background: 'hsl(var(--background))',
* foreground: 'hsl(var(--foreground))',
* card: {
* DEFAULT: 'hsl(var(--card))',
* foreground: 'hsl(var(--card-foreground))',
* },
* popover: {
* DEFAULT: 'hsl(var(--popover))',
* foreground: 'hsl(var(--popover-foreground))',
* },
* primary: {
* DEFAULT: 'hsl(var(--primary))',
* foreground: 'hsl(var(--primary-foreground))',
* },
* secondary: {
* DEFAULT: 'hsl(var(--secondary))',
* foreground: 'hsl(var(--secondary-foreground))',
* },
* muted: {
* DEFAULT: 'hsl(var(--muted))',
* foreground: 'hsl(var(--muted-foreground))',
* },
* accent: {
* DEFAULT: 'hsl(var(--accent))',
* foreground: 'hsl(var(--accent-foreground))',
* },
* destructive: {
* DEFAULT: 'hsl(var(--destructive))',
* foreground: 'hsl(var(--destructive-foreground))',
* },
* border: 'hsl(var(--border))',
* input: 'hsl(var(--input))',
* ring: 'hsl(var(--ring))',
* chart: {
* 1: 'hsl(var(--chart-1))',
* 2: 'hsl(var(--chart-2))',
* 3: 'hsl(var(--chart-3))',
* 4: 'hsl(var(--chart-4))',
* 5: 'hsl(var(--chart-5))',
* },
* },
* },
* },
* plugins: [require('tailwindcss-animate')],
* };
* Tsconfig.json
* {
* "compilerOptions": {
* "target": "esnext",
* "lib": ["dom", "dom.iterable", "esnext"],
* "jsx": "preserve",
* "strict": false,
* "esModuleInterop": true,
* "module": "esnext",
* "moduleResolution": "node",
* "allowJs": true,
* "skipLibCheck": true,
* "forceConsistentCasingInFileNames": true,
* "resolveJsonModule": true,
* "isolatedModules": true,
* "noEmit": true,
* "allowImportingTsExtensions": true,
* "baseUrl": ".",
* "paths": {
* "@/\*": ["./client/src/\*"]
* }
* }
* }
* Tsconfig.server.json
* {
* "compilerOptions": {
* "target": "es2020",
* "module": "NodeNext",
* "moduleResolution": "NodeNext",
* "esModuleInterop": true,
* "outDir": "dist/server",
* "skipLibCheck": true,
* "forceConsistentCasingInFileNames": true,
* "resolveJsonModule": true
* },
* "include": ["server/\*\*/\*"],
* "exclude": ["node\_modules"]
* }
* [vite.config.js](http://vite.config.js)

import { defineConfig } from 'vite';

import react from '@vitejs/plugin-react';

import path from 'path';

export const vitePort = 3000;

export default defineConfig(({ mode }) => {

return {

plugins: [

react(),

// Custom plugin to handle source map requests

{

name: 'handle-source-map-requests',

apply: 'serve',

configureServer(server) {

server.middlewares.use((req, res, next) => {

// Check if the request is for a source map file

if (req.url && req.url.endsWith('.map')) {

// Rewrite the URL to remove the query string that's causing the issue

const cleanUrl = req.url.split('?')[0];

req.url = cleanUrl;

}

next();

});

},

},

// Custom plugin to add CORS headers

{

name: 'add-cors-headers',

apply: 'serve',

configureServer(server) {

server.middlewares.use((req, res, next) => {

// Add CORS headers to all responses

res.setHeader('Access-Control-Allow-Origin', '\*');

res.setHeader(

'Access-Control-Allow-Methods',

'GET, POST, PUT, DELETE, PATCH, OPTIONS',

);

res.setHeader(

'Access-Control-Allow-Headers',

'Content-Type, Authorization, X-Requested-With',

);

// Handle OPTIONS requests

if (req.method === 'OPTIONS') {

res.statusCode = 204;

return res.end();

}

next();

});

},

},

].filter(Boolean),

resolve: {

alias: {

'@': path.resolve(\_\_dirname, './client/src'),

},

},

root: path.join(process.cwd(), 'client'),

build: {

outDir: path.join(process.cwd(), 'dist/public'),

emptyOutDir: true,

},

clearScreen: false,

server: {

hmr: {

overlay: false,

},

host: true,

port: vitePort,

allowedHosts: true,

cors: true, // Enable CORS in the dev server

proxy: {

'/api/': {

target: 'http://localhost:3001',

changeOrigin: true,

},

},

},

// Enable source maps for development

css: {

devSourcemap: true,

},

// Ensure source maps are properly generated

esbuild: {

sourcemap: true,

},

};

});