

## index.js

```
import React from "react";
import ReactDOM from "react-dom";
import "./index.css";
import App from "./App";
import "react-alice-carousel/lib/alice-carousel.css";
import CryptoContext from "./CryptoContext";

ReactDOM.render(
  <React.StrictMode>
    <CryptoContext>
      <App />
    </CryptoContext>
  </React.StrictMode>,
  document.getElementById("root")
);
```

## index.css

```
body {
  margin: 0;
  font-family: -apple-system, BlinkMacSystemFont, 'Segoe UI', 'Roboto', 'Oxygen',
    'Ubuntu', 'Cantarell', 'Fira Sans', 'Droid Sans', 'Helvetica Neue',
    sans-serif;
  -webkit-font-smoothing: antialiased;
  -moz-osx-font-smoothing: grayscale;
}

code {
  font-family: source-code-pro, Menlo, Monaco, Consolas, 'Courier New',
    monospace;
}
```









## CryptoContext.js

```
import React, { createContext, useContext, useEffect, useState } from 'react';
const Crypto = createContext();

const CryptoContext = ({children}) => {
  const [currency, setCurrency] = useState("USD");
  const [symbol, setSymbol] = useState("$");

  useEffect(() => {
    if (currency === "USD") setSymbol("$");
    else if (currency === "GBP") setSymbol("£");
    else if (currency === "EUR") setSymbol("€");
    else if (currency === "AUD") setSymbol("AU$");
    else if (currency === "CAD") setSymbol("CA$");
    else if (currency === "INR") setSymbol("₹");
  }, [currency]);

  return (
    <Crypto.Provider value={{currency,symbol,setCurrency}}>{children}</Crypto.Provider>
  )
};
export default CryptoContext;
export const CryptoState = () => {
  return useContext(Crypto);
}
```

	Pages
	components
	config
	App.css
	App.js
	CryptoContext.js
	index.css
	index.js









## App.js

```
import { makeStyles } from "@material-ui/core";
import Homepage from "../Pages/HomePage";
import "../App.css";
import { BrowserRouter, Route } from "react-router-dom";
import CoinPage from "../Pages/CoinPage";
import Header from "../components/Header";

const useStyles = makeStyles() => ({
  App: {
    backgroundColor: "#14161a",
    color: "white",
    minHeight: "100vh",
  },
});

function App() {
  const classes = useStyles();

  return (
    <BrowserRouter>
    <div className={classes.App}>
      <Header />
      <Route path="/" component={Homepage} exact />
      <Route path="/coins/:id" component={CoinPage} exact />
    </div>
    </BrowserRouter >
  );
}
export default App;
```

	Pages
	components
	config
	App.css
	App.js
	CryptoContext.js
	index.css
	index.js

## App.css

```
@import url("https://fonts.googleapis.com/css2?family=Montserrat:wght@300;400;800&display=swap");

* {
  margin: 0;
  padding: 0;
  box-sizing: border-box;
  scroll-behavior: smooth;
}

a {
  text-decoration: none;
  color: gold;
}
```

## CONFIG

config/data.js  
config/api.js

### **config/data.js**

```
export const chartDays = [
  {
    label: "24 Hours",
    value: 1,
  },
  {
    label: "30 Days",
    value: 30,
  },
  {
    label: "3 Months",
    value: 90,
  },
  {
    label: "1 Year",
    value: 365,
  },
];
```









### **config/api.js**

```
export const CoinList = (currency) =>
`https://api.coingecko.com/api/v3/coins/markets?vs_currency=$
{currency}&order=market_cap_desc&per_page=100&page=1&sparkline=false`;

export const SingleCoin = (id) =>
`https://api.coingecko.com/api/v3/coins/${id}`;

export const HistoricalChart = (id, days = 365, currency) =>
`https://api.coingecko.com/api/v3/coins/${id}/market_chart?vs_currency=${currency}&days=${days}`;

export const TrendingCoins = (currency) =>
`https://api.coingecko.com/api/v3/coins/markets?vs_currency=$
{currency}&order=gecko_desc&per_page=10&page=1&sparkline=false&price_change_percentage=24h`;
```

	Pages
	components
	config
	App.css
	App.js
	CryptoContext.js
	index.css
	index.js

## COMPONENTS

components/Banner/Carousel.js  
components/Banner/Banner.js  
components/CoinInfo.js  
components/CoinTable.js  
components/Header.js  
components/SelectButton.js

### **components/Banner/Carousel.js**

```
import { makeStyles } from "@material-ui/core";
import axios from "axios";
import { useEffect, useState } from "react";
import AliceCarousel from "react-alice-carousel";
import { Link } from "react-router-dom";
import { TrendingCoins } from "../../config/api";
import { CryptoState } from "../../CryptoContext";
import { numberWithCommas } from "../../CoinsTable";

const Carousel = () => {
  const [trending, setTrending] = useState([]);
  const { currency, symbol } = CryptoState();

  const fetchTrendingCoins = async () => {
    const { data } = await axios.get(TrendingCoins(currency));









    console.log(data);
    setTrending(data);
  };

  useEffect(() => {
    fetchTrendingCoins();
    // eslint-disable-next-line react-hooks/exhaustive-deps
  }, [currency]);

  const useStyles = makeStyles((theme) => ({
    carousel: {
      height: "50%",
      display: "flex",
      alignItems: "center",
    },
    carouselItem: {
      display: "flex",
      flexDirection: "column",
      alignItems: "center",
      cursor: "pointer",
      textTransform: "uppercase",
      color: "white",
    },
  }));

  const classes = useStyles();

  const items = trending.map((coin) => {
    let profit = coin?.price_change_percentage_24h >= 0;
```

	Pages
	components
	config
	App.css
	App.js
	CryptoContext.js
	index.css
	index.js

```

return (
  <Link className={classes.carouselItem} to={` /coins/${coin.id}`} >
    <img
      src={coin?.image}
      alt={coin.name}
      height="80"
      style={{ marginBottom: 10 }}
    />
    <span>
      {coin?.symbol}
      &nbsp;
      <span
        style={{
          color: profit > 0 ? "rgb(14, 203, 129)" : "red",
          fontWeight: 500,
        }}
      >
        {profit && "+"}
        {coin?.price_change_percentage_24h?.toFixed(2)}%
      </span>
    </span>
    <span style={{ fontSize: 22, fontWeight: 500 }} >
      {symbol} {numberWithCommas(coin?.current_price.toFixed(2))}
    </span>
  </Link>
);
});

```

```

const responsive = {
  0: {
    items: 2,
  },
  512: {
    items: 4,
  },
};

```

```

return (
  <div className={classes.carousel}>
    <AliceCarousel
      mouseTracking
      infinite
      autoPlayInterval={1000}
      animationDuration={1500}
      disableDotsControls
      disableButtonsControls
      responsive={responsive}
      items={items}
      autoPlay
    />
  </div>
);
};

```

```

export default Carousel;

```

## components/Banner/Banner.js

```
import { Container, makeStyles, Typography } from "@material-ui/core";
import Carousel from "../Carousel";
```

```
const useStyles = makeStyles((theme) => ({
  banner: {
    backgroundImage: "url(/banner3.jpg)",
  },
  bannerContent: {
    height: 400,
    display: "flex",
    flexDirection: "column",
    paddingTop: 25,
    justifyContent: "space-around",
  },
  tagline: {
    display: "flex",
    height: "40%",
    flexDirection: "column",
    justifyContent: "center",
    textAlign: "center",
  },
  carousel: {
    height: "50%",
    display: "flex",
    alignItems: "center",
  },
}));
```

```
function Banner() {
  const classes = useStyles();
  return (
    <div className={classes.banner}>
      <Container className={classes.bannerContent}>
        <div className={classes.tagline}>
          <Typography
            variant="h2"
            style={{
              fontWeight: "bold",
              marginBottom: 15,
              fontFamily: "Montserrat",
            }}
          >
            Crypto Tracker
          </Typography>
          <Typography
            variant="subtitle2"
            style={{
              color: "darkgrey",
              textTransform: "capitalize",
              fontFamily: "Montserrat",
            }}
          >
            Get all the Info regarding your favorite Crypto Currency
          </Typography>
        </div>
        <Carousel />
      </Container>
    </div>
  ); }
export default Banner;
```

## components/CoinInfo.js

```
import axios from "axios";
import { useEffect, useState } from "react";
import { HistoricalChart } from "../config/api";
import { Line } from "react-chartjs-2";
import {
  CircularProgress,
  createTheme,
  makeStyles,
  ThemeProvider,
} from "@material-ui/core";
import SelectButton from "../SelectButton";
import { chartDays } from "../config/data";
import { CryptoState } from "../CryptoContext";

const CoinInfo = ({ coin }) => {
  const [historicData, setHistoricData] = useState();
  const [days, setDays] = useState(1);
  const { currency } = CryptoState();
  const [flag, setflag] = useState(false);

  const useStyles = makeStyles((theme) => ({
    container: {
      width: "75%",
      display: "flex",
      flexDirection: "column",
      alignItems: "center",
      justifyContent: "center",
      marginTop: 25,
      padding: 40,
      [theme.breakpoints.down("md")]: {
        width: "100%",
        marginTop: 0,
        padding: 20,
        paddingTop: 0,
      },
    },
  }));

  const classes = useStyles();

  const fetchHistoricData = async () => {
    const { data } = await axios.get(HistoricalChart(coin.id, days, currency));
    setflag(true);
    setHistoricData(data.prices);
  };

  useEffect(() => {
    fetchHistoricData();
    // eslint-disable-next-line react-hooks/exhaustive-deps
  }, [days]);

  const darkTheme = createTheme({
    palette: {
      primary: {
        main: "#fff",
      },
      type: "dark",
    },
  });
```

```

return (
  <ThemeProvider theme={darkTheme}>
    <div className={classes.container}>
      {!historicData | flag===false ? (
        <CircularProgress
          style={{ color: "gold" }}
          size={250}
          thickness={1}
        />
      ) : (
        <>
          <Line
            data={{
              labels: historicData.map((coin) => {
                let date = new Date(coin[0]);
                let time =
                  date.getHours() > 12
                    ? `${date.getHours() - 12}:${date.getMinutes()} PM`
                    : `${date.getHours()}:${date.getMinutes()} AM`;
                return days === 1 ? time : date.toLocaleDateString();
              }),

              datasets: [
                {
                  data: historicData.map((coin) => coin[1]),
                  label: `Price ( Past ${days} Days ) in ${currency}`,
                  borderColor: "#EEBC1D",
                },
              ],
            }}
            options={{
              elements: {
                point: {
                  radius: 1,
                },
              },
            }}
          />
        <div
          style={{
            display: "flex",
            marginTop: 20,
            justifyContent: "space-around",
            width: "100%",
          }}
        >
          {chartDays.map((day) => (
            <SelectButton
              key={day.value}
              onClick={() => {setDays(day.value);
                setflag(false);
              }}
              selected={day.value === days}
            >
              {day.label}
            </SelectButton>
          ))}
        </div>
      </>
    )}
  </div>
</ThemeProvider>
);
};
export default CoinInfo;

```



## components/CoinTable.js

```
import React, { useEffect, useState } from "react";
import { makeStyles } from "@material-ui/core/styles";
import Pagination from "@material-ui/lab/Pagination";
import { Container, createTheme, TableCell, LinearProgress, ThemeProvider, Typography, TextField, TableBody,
  TableRow, TableHead, TableContainer, Table, Paper } from "@material-ui/core";
import axios from "axios";
import { CoinList } from "../config/api";
import { useHistory } from "react-router-dom";
import { CryptoState } from "../CryptoContext";

export function numberWithCommas(x) {
  return x.toString().replace(/\B(?=(\d{3})+(?!\d))/g, ",");
}

export default function CoinsTable() {
  const [coins, setCoins] = useState([]);
  const [loading, setLoading] = useState(false);
  const [search, setSearch] = useState("");
  const [page, setPage] = useState(1);

  const { currency, symbol } = CryptoState();

  const useStyles = makeStyles({
    row: {
      backgroundColor: "#16171a",
      cursor: "pointer",
      "&:hover": {
        backgroundColor: "#131111",
      },
      fontFamily: "Montserrat",
    },
    pagination: {
      "& .MuiPaginationItem-root": {
        color: "gold",
      },
    },
  });

  const classes = useStyles();
  const history = useHistory();

  const darkTheme = createTheme({
    palette: {
      primary: {
        main: "#fff",
      },
      type: "dark",
    },
  });

  const fetchCoins = async () => {
    setLoading(true);
    const { data } = await axios.get(CoinList(currency));
    console.log(data);

    setCoins(data);
    setLoading(false);
  };
}
```

```

useEffect(() => {
  fetchCoins();
  // eslint-disable-next-line react-hooks/exhaustive-deps
}, [currency]);

const handleSearch = () => {
  return coins.filter(
    (coin) =>
      coin.name.toLowerCase().includes(search) ||
      coin.symbol.toLowerCase().includes(search)
  );
};

return (
  <ThemeProvider theme={darkTheme}>
    <Container style={{ textAlign: "center" }}>
      <Typography
        variant="h4"
        style={{ margin: 18, fontFamily: "Montserrat" }}
      >
        Cryptocurrency Prices by Market Cap
      </Typography>

      <TextField
        label="Search For a Crypto Currency.."
        variant="outlined"
        style={{ marginBottom: 20, width: "100%" }}
        onChange={(e) => setSearch(e.target.value)}
      />

      <TableContainer component={Paper}>

        {loading ? (
          <LinearProgress style={{ backgroundColor: "gold" }} />
        ) : (

          <Table aria-label="simple table">

            <TableHead style={{ backgroundColor: "#EEBC1D" }}>
              <TableRow>
                {["Coin", "Price", "24h Change", "Market Cap"].map((head) => (
                  <TableCell
                    style={{
                      color: "black",
                      fontWeight: "700",
                      fontFamily: "Montserrat",
                    }}
                    key={head}
                    align={head === "Coin" ? "" : "right"}
                  >
                    {head}
                  </TableCell>
                ))}
              </TableRow>
            </TableHead>

```

```

<TableBody>
{handleSearch()
.slice((page - 1) * 10, (page - 1) * 10 + 10)
.map((row) => {
  const profit = row.price_change_percentage_24h > 0;
  return (
    <TableRow
      onClick={() => history.push(`/coins/${row.id}`)}
      className={classes.row}
      key={row.name}
    >
      <TableCell
        component="th"
        scope="row"
        style={{
          display: "flex",
          gap: 15,
        }}
      >
        <img
          src={row?.image}
          alt={row.name}
          height="50"
          style={{ marginBottom: 10 }}
        />
        <div
          style={{ display: "flex", flexDirection: "column" }}
        >
          <span
            style={{
              textTransform: "uppercase",
              fontSize: 22,
            }}
          >
            {row.symbol}
          </span>
          <span style={{ color: "darkgrey" }}>
            {row.name}
          </span>
        </div>
      </TableCell>
      <TableCell align="right">
        {symbol}{" "}
        {numberWithCommas(row.current_price.toFixed(2))}
      </TableCell>
      <TableCell
        align="right"
        style={{
          color: profit > 0 ? "rgb(14, 203, 129)" : "red",
          fontWeight: 500,
        }}
      >
        {profit && "+"}
        {row.price_change_percentage_24h.toFixed(2)}%
      </TableCell>
      <TableCell align="right">
        {symbol}{" "}
        {numberWithCommas(
          row.market_cap.toString().slice(0, -6)
        )}
        M
      </TableCell>
    </TableRow>
  )
}
)
}

```

```

    );
  })}
</TableBody>

</Table>
)}
</TableContainer>

{/* Comes from @material-ui/lab */}
<Pagination
  count={(handleSearch()?.length / 10).toFixed(0)}
  style={{
    padding: 20,
    width: "100%",
    display: "flex",
    justifyContent: "center",
  }}
  classes={{ ul: classes.pagination }}
  onChange={(_, value) => {
    setPage(value);
    window.scroll(0, 450);
  }}
/>
</Container>
</ThemeProvider>
);
}

```

## components/Header.js

```

import {
  AppBar,
  Container,
  MenuItem,
  Select,
  Toolbar,
  Typography,
} from "@material-ui/core";
import {
  createTheme,
  makeStyles,
  ThemeProvider,
} from "@material-ui/core/styles";
import { useHistory } from "react-router-dom";
import { CryptoState } from "../CryptoContext";

const useStyles = makeStyles((theme) => ({
  title: {
    flex: 1,
    color: "gold",
    fontFamily: "Montserrat",
    fontWeight: "bold",
    cursor: "pointer",
  },
}));

```

```

const darkTheme = createTheme({
  palette: {
    primary: {
      main: "#fff",
    },
    type: "dark",
  },
});

function Header() {
  const classes = useStyles();
  const { currency, setCurrency } = CryptoState();

  const history = useHistory();

  return (
    <ThemeProvider theme={darkTheme}>
      <AppBar color="transparent" position="static">
        <Container>
          <Toolbar>
            <Typography
              onClick={() => history.push('/')}
              variant="h6"
              className={classes.title}
            >
              Crypto Tracker
            </Typography>
            { /* <Button color="inherit">Login</Button> */ }
            <Select
              variant="outlined"
              labelId="demo-simple-select-label"
              id="demo-simple-select"
              value={currency}
              style={{ width: 100, height: 40, marginLeft: 15 }}
              onChange={(e) => setCurrency(e.target.value)}
            >
              <MenuItem value={"USD"}>USD</MenuItem>
              <MenuItem value={"GBP"}>GBP</MenuItem>
              <MenuItem value={"EUR"}>EUR</MenuItem>
              <MenuItem value={"AUD"}>AUD</MenuItem>
              <MenuItem value={"CAD"}>CAD</MenuItem>
              <MenuItem value={"INR"}>INR</MenuItem>

            </Select>
          </Toolbar>
        </Container>
      </AppBar>
    </ThemeProvider>
  );
}

export default Header;

```

## components/SelectBUtton.js

```
import { makeStyles } from "@material-ui/core";

const SelectButton = ({ children, selected, onClick }) => {
  const useStyles = makeStyles({
    selectbutton: {
      border: "1px solid gold",
      borderRadius: 5,
      padding: 10,
      paddingLeft: 20,
      paddingRight: 20,
      fontFamily: "Montserrat",
      cursor: "pointer",
      backgroundColor: selected ? "gold" : "",
      color: selected ? "black" : "",
      fontWeight: selected ? 700 : 500,
      "&:hover": {
        backgroundColor: "gold",
        color: "black",
      },
      width: "22%",
      // margin: 5,
    },
  });

  const classes = useStyles();

  return (
    <span onClick={onClick} className={classes.selectbutton}>
      {children}
    </span>
  );
};

export default SelectButton;
```

## PAGES

Pages/HomePage.js  
Pages/CoinPage.js

### Pages/HompePage.js

```
import React from "react";
import Banner from "../components/Banner/Banner";
import CoinsTable from "../components/CoinsTable";

const Homepage = () => {
  return (
    <>
      <Banner />
      <CoinsTable />
    </>
  );
};

export default Homepage;
```

### Pages/CoinPage.js

```
import { LinearProgress, makeStyles, Typography } from "@material-ui/core";
import axios from "axios";
import { useEffect, useState } from "react";
import { useParams } from "react-router-dom";
import ReactHtmlParser from "react-html-parser";
import CoinInfo from "../components/CoinInfo";
import { SingleCoin } from "../config/api";
import { numberWithCommas } from "../components/CoinsTable";
import { CryptoState } from "../CryptoContext";

const CoinPage = () => {
  const { id } = useParams();
  const [coin, setCoin] = useState();









  const { currency, symbol } = CryptoState();

  const fetchCoin = async () => {
    const { data } = await axios.get(SingleCoin(id));

    setCoin(data);
  };

  useEffect(() => {
    fetchCoin();
    // eslint-disable-next-line react-hooks/exhaustive-deps
  }, []);

  const useStyles = makeStyles((theme) => ({
    container: {
      display: "flex",
      [theme.breakpoints.down("md")]: {
        flexDirection: "column",
        alignItems: "center",
      },
    },
  }));
```

	Pages
	components
	config
	App.css
	App.js
	CryptoContext.js
	index.css
	index.js

```

},
sidebar: {
  width: "30%",
  [theme.breakpoints.down("md")]: {
    width: "100%",
  },
  display: "flex",
  flexDirection: "column",
  alignItems: "center",
  marginTop: 25,
  borderRight: "2px solid grey",
},
heading: {
  fontWeight: "bold",
  marginBottom: 20,
  fontFamily: "Montserrat",
},
description: {
  width: "100%",
  fontFamily: "Montserrat",
  padding: 25,
  paddingBottom: 15,
  paddingTop: 0,
  textAlign: "justify",
},
marketData: {
  alignSelf: "start",
  padding: 25,
  paddingTop: 10,
  width: "100%",
  [theme.breakpoints.down("md")]: {
    display: "flex",
    justifyContent: "space-around",
  },
  [theme.breakpoints.down("sm")]: {
    flexDirection: "column",
    alignItems: "center",
  },
  [theme.breakpoints.down("xs")]: {
    alignItems: "start",
  },
},
}));

```

```
const classes = useStyles();
```

```
if (!coin) return <LinearProgress style={{ backgroundColor: "gold" }} />;
```

```

return (
  <div className={classes.container}>
    <div className={classes.sidebar}>
      <img
        src={coin?.image.large}
        alt={coin?.name}
        height="200"
        style={{ marginBottom: 20 }}
      />
      <Typography variant="h3" className={classes.heading}>
        {coin?.name}
      </Typography>
      <Typography variant="subtitle1" className={classes.description}>
        {ReactHtmlParser(coin?.description.en.split(". ")[0])}.
      </Typography>
    </div>
  </div>
)

```



```

<div className={classes.marketData}>
  <span style={{ display: "flex" }}>
    <Typography variant="h5" className={classes.heading}>
      Rank:
    </Typography>
    &nbsp; &nbsp;
    <Typography
      variant="h5"
      style={{
        fontFamily: "Montserrat",
      }}
    >
      {numberWithCommas(coin?.market_cap_rank)}
    </Typography>
  </span>

  <span style={{ display: "flex" }}>
    <Typography variant="h5" className={classes.heading}>
      Current Price:
    </Typography>
    &nbsp; &nbsp;
    <Typography
      variant="h5"
      style={{
        fontFamily: "Montserrat",
      }}
    >
      {symbol}{" "}
      {numberWithCommas(
        coin?.market_data.current_price[currency.toLowerCase()]
      )}
    </Typography>
  </span>

  <span style={{ display: "flex" }}>
    <Typography variant="h5" className={classes.heading}>
      Market Cap:
    </Typography>
    &nbsp; &nbsp;
    <Typography
      variant="h5"
      style={{
        fontFamily: "Montserrat",
      }}
    >
      {symbol}{" "}
      {numberWithCommas(
        coin?.market_data.market_cap[currency.toLowerCase()]
          .toString()
          .slice(0, -6)
      )}
      M
    </Typography>
  </span>
</div>
</div>
<CoinInfo coin={coin} />
</div>
);
};

```

```
export default CoinPage;
```

## package.json

```
{
  "name": "crypto-tracker",
  "version": "0.1.0",
  "private": true,

  "dependencies": {
    "@material-ui/core": "4.12.3",
    "@material-ui/lab": "^4.0.0-alpha.60",

    "@testing-library/jest-dom": "^5.14.1",
    "@testing-library/react": "^12.0.0",
    "@testing-library/user-event": "^13.2.1",

    "axios": "^0.21.1",

    "chart.js": "^3.5.1",

    "react-chartjs-2": "^3.0.4",
    "react": "^17.0.2",
    "react-alice-carousel": "^2.5.1",
    "react-dom": "^17.0.2",
    "react-html-parser": "^2.0.2",
    "react-router-dom": "^5.2.0",
    "react-scripts": "4.0.3",

    "web-vitals": "^1.0.1"
  },

  "scripts": {
    "start": "export SET NODE_OPTIONS=--openssl-legacy-provider && react-scripts start",
    "build": "export SET NODE_OPTIONS=--openssl-legacy-provider && react-scripts build",
    "test": "react-scripts test",
    "eject": "react-scripts eject"
  },

  "eslintConfig": {
    "extends": [
      "react-app",
      "react-app/jest"
    ]
  },

  "browserslist": {
    "production": [
      ">0.2%",
      "not dead",
      "not op_mini all"
    ],
    "development": [
      "last 1 chrome version",
      "last 1 firefox version",
      "last 1 safari version"
    ]
  }
}
```

**public**

public/banner3.jpg

**cmd: “yarn start” <Enter>**