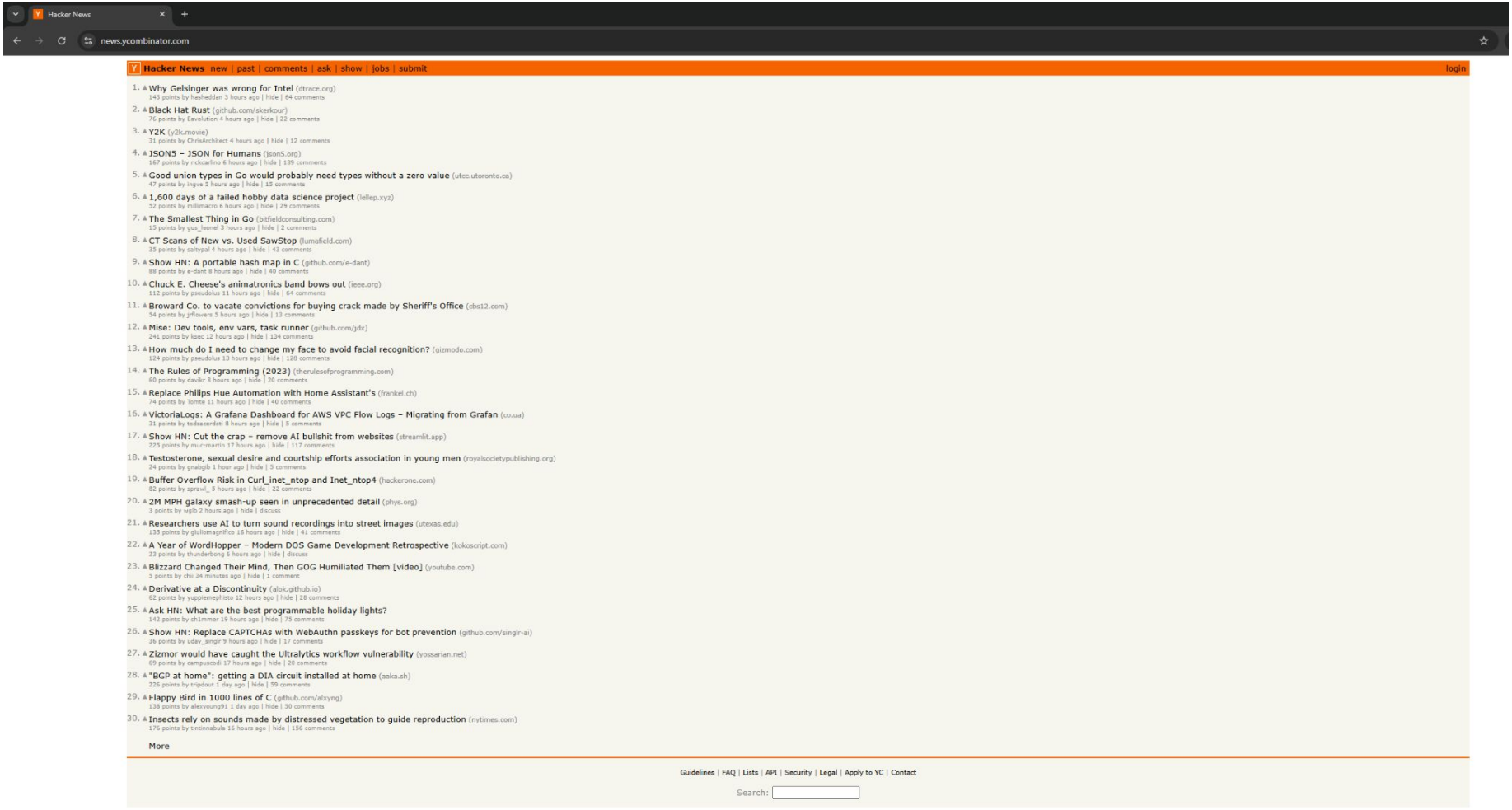
HN DB

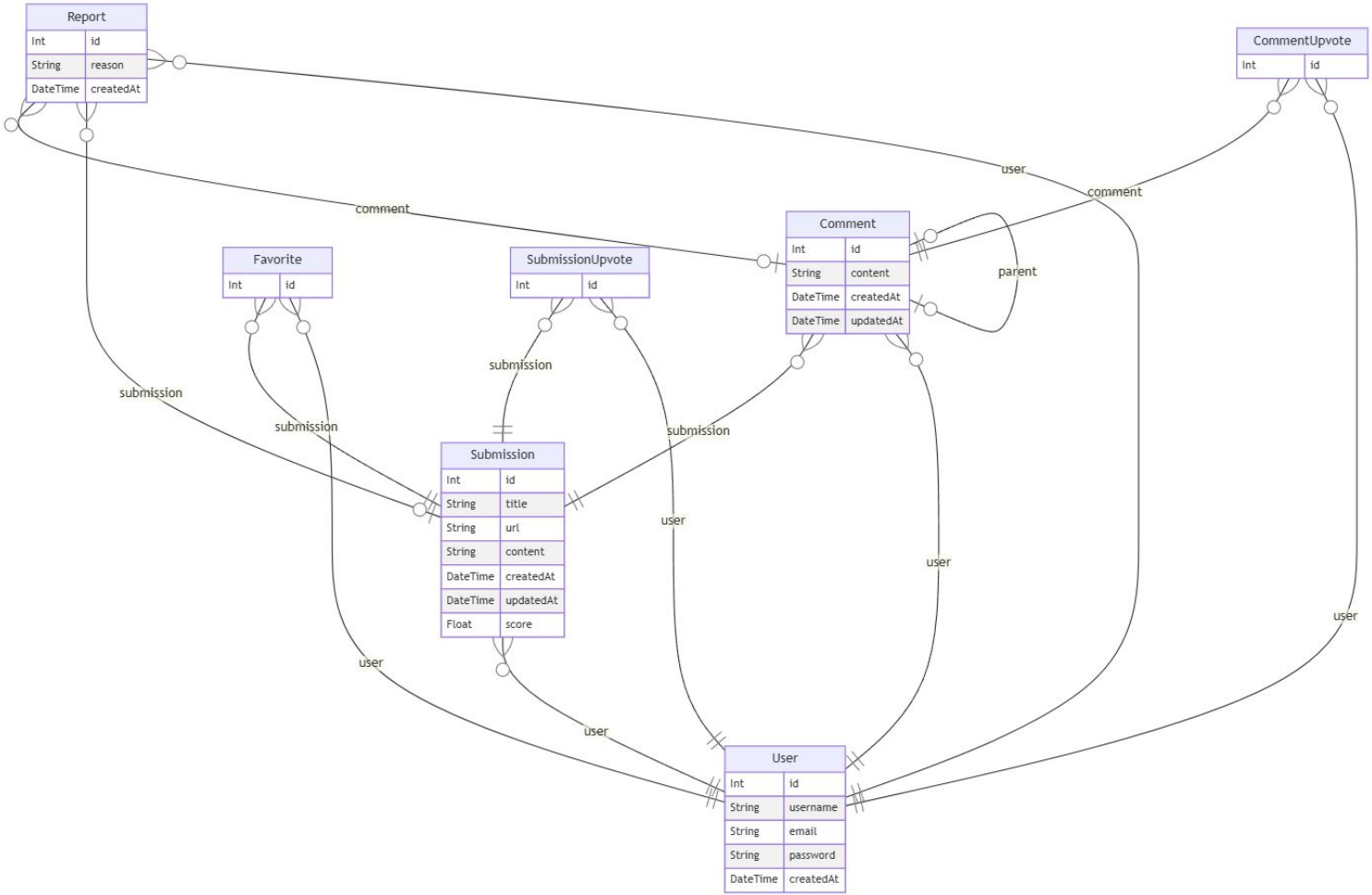
Jacob Hummer

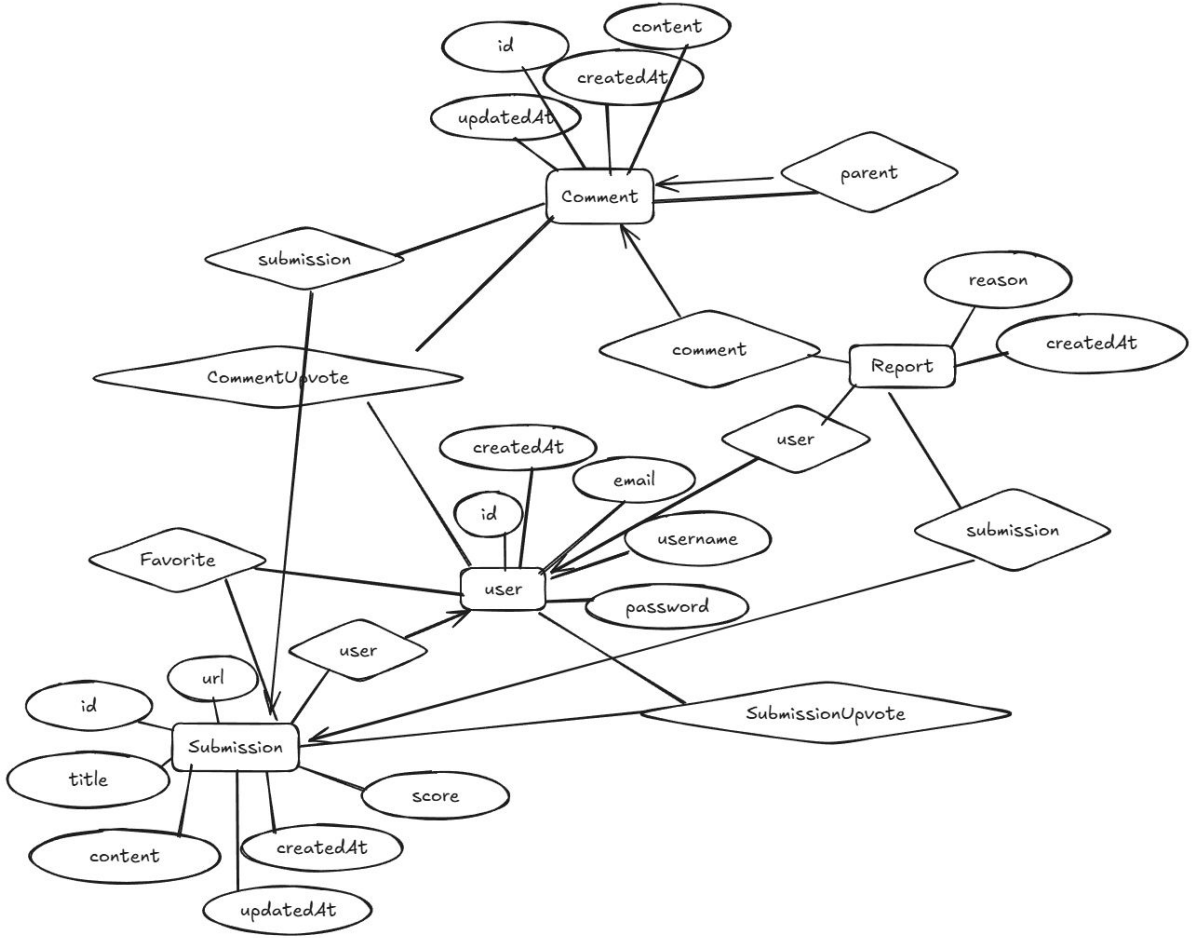
Description


Hacker News
news.ycombinator.com
Hacker News new | past | comments | ask | show | jobs | submit login
1. Why Gelsinger was wrong for Intel (dtrace.org)
143 points by hasheddan 3 hours ago | hide | 64 comments
2. Black Hat Rust (github.com/skerkour)
76 points by Eavolution 4 hours ago | hide | 22 comments
3. Y2K (y2k.movie)
31 points by ChrisArchitect 4 hours ago | hide | 12 comments
4. JSON5 – JSON for Humans (json5.org)
167 points by rickcarlino 6 hours ago | hide | 139 comments
5. Good union types in Go would probably need types without a zero value (utcc.utoronto.ca)
47 points by ingve 5 hours ago | hide | 15 comments
6. 1,600 days of a failed hobby data science project (lellep.xyz)
52 points by millimacro 6 hours ago | hide | 29 comments
7. The Smallest Thing in Go (bitfieldconsulting.com)
15 points by gus_leonel 3 hours ago | hide | 2 comments
8. CT Scans of New vs. Used SawStop (lumafield.com)
35 points by saltypal 4 hours ago | hide | 43 comments
9. Show HN: A portable hash map in C (github.com/e-dant)
88 points by e-dant 8 hours ago | hide | 40 comments
10. Chuck E. Cheese's animatronics band bows out (ieee.org)
112 points by pseudolus 11 hours ago | hide | 64 comments
11. Broward Co. to vacate convictions for buying crack made by Sheriff's Office (cbs12.com)
54 points by jrflowers 5 hours ago | hide | 13 comments
12. Mise: Dev tools, env vars, task runner (github.com/jdx)
241 points by ksec 12 hours ago | hide | 134 comments
13. How much do I need to change my face to avoid facial recognition? (gizmodo.com)
124 points by pseudolus 13 hours ago | hide | 128 comments
14. The Rules of Programming (2023) (therulesofprogramming.com)
60 points by davikr 8 hours ago | hide | 20 comments
15. Replace Philips Hue Automation with Home Assistant's (frankel.ch)
74 points by Tomte 11 hours ago | hide | 40 comments
16. VictoriaLogs: A Grafana Dashboard for AWS VPC Flow Logs – Migrating from Grafan (co.ua)
31 points by todbeerdeti 8 hours ago | hide | 5 comments
17. Show HN: Cut the crap – remove AI bullshit from websites (streamlit.app)
223 points by mac-martin 17 hours ago | hide | 117 comments
18. Testosterone, sexual desire and courtship efforts association in young men (royalsocietypublishing.org)
24 points by gnabgib 1 hour ago | hide | 5 comments
19. Buffer Overflow Risk in Curl_inet_ntop and Inet_ntop4 (hackerone.com)
82 points by sprawl_ 5 hours ago | hide | 22 comments
20. 2M MPH galaxy smash-up seen in unprecedented detail (phys.org)
3 points by wglb 2 hours ago | hide | discuss
21. Researchers use AI to turn sound recordings into street images (utexas.edu)
135 points by giuliomagnifico 16 hours ago | hide | 41 comments
22. A Year of WordHopper – Modern DOS Game Development Retrospective (kokoscript.com)
23 points by thunderbong 6 hours ago | hide | discuss
23. Blizzard Changed Their Mind, Then GOG Humiliated Them [video] (youtube.com)
9 points by chii 34 minutes ago | hide | 1 comment
24. Derivative at a Discontinuity (alok.github.io)
62 points by yuppiemephisto 12 hours ago | hide | 28 comments
25. Ask HN: What are the best programmable holiday lights?
142 points by sh1mmer 19 hours ago | hide | 75 comments
26. Show HN: Replace CAPTCHAs with WebAuthn passkeys for bot prevention (github.com/singlr-ai)
36 points by udav_singlr 9 hours ago | hide | 17 comments
27. Zizmor would have caught the Ultralytics workflow vulnerability (yossarian.net)
69 points by campuscodi 17 hours ago | hide | 20 comments
28. "BGP at home": getting a DIA circuit installed at home (aaka.sh)
226 points by tripdout 1 day ago | hide | 59 comments
29. Flappy Bird in 1000 lines of C (github.com/alxyng)
138 points by alexyoung91 1 day ago | hide | 50 comments
30. Insects rely on sounds made by distressed vegetation to guide reproduction (nytimes.com)
176 points by tintinnabula 16 hours ago | hide | 156 comments
More
Guidelines | FAQ | Lists | API | Security | Legal | Apply to YC | Contact
Search:

- **Users**
- **Submissions**
- **Comments**
- Reports
- Favorites

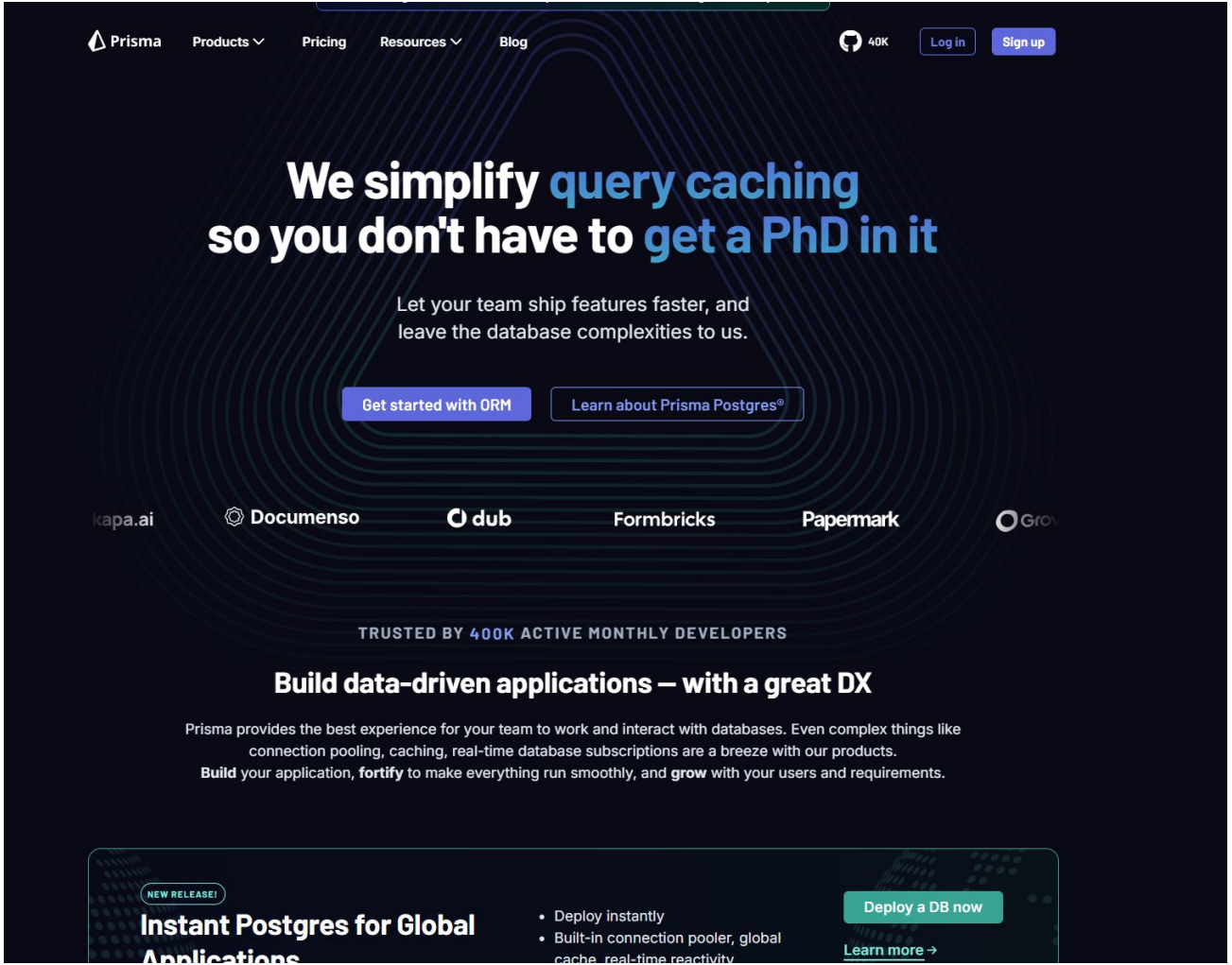
Implementation


Report
Int id
String reason
DateTime createdAt
CommentUpvote
Int id
Favorite
Int id
SubmissionUpvote
Int id
Comment
Int id
String content
DateTime createdAt
DateTime updatedAt
Submission
Int id
String title
String url
String content
DateTime createdAt
DateTime updatedAt
Float score
User
Int id
String username
String email
String password
DateTime createdAt
user
comment
comment
parent
submission
submission
submission
submission
user
user
user
user
user
user


id
content
createdAt
updatedAt
Comment
parent
submission
reason
createdAt
comment
Report
CommentUpvote
user
createdAt
email
id
username
submission
Favorite
user
password
user
url
id
SubmissionUpvote
Submission
title
score
content
createdAt
updatedAt

```
}

datasource db {
  provider  = "postgresql"
  url       = env("POSTGRES_PRISMA_URL") // uses connection pooling
  directUrl = env("POSTGRES_URL_NON_POOLING") // uses a direct connection
}

model User {
  id                 Int               @id @default(autoincrement())
  username           String            @unique
  email              String            @unique
  password           String
  createdAt          DateTime          @default(now())
  submissions        Submission[]
  comments           Comment[]
  favorites          Favorite[]
  upvotedSubmissions SubmissionUpvote[]
  upvotedComments    CommentUpvote[]
  reports            Report[]
}

// The score field in the Submission model will be updated using a scheduled job in JavaScript
// to decay over time. The score is calculated based on the number of upvotes and the age of the submission.
model Submission {
  id        Int               @id @default(autoincrement())
  title     String
  url       String?
  content   String?
  createdAt DateTime          @default(now())
  updatedAt DateTime          @updatedAt
  userId    Int
  user      User              @relation(fields: [userId], references: [id])
  comments  Comment[]
  upvotes   SubmissionUpvote[]
  score     Float             @default(0) // Multiplied by 0.5 every cron "0 5 * * *" by /api/decay-submission-score
  Favorite  Favorite[]
  Report    Report[]
}

model Comment {
  id           Int            @id @default(autoincrement())
```

https://www.prisma.io/



```
const user = await prisma.user.create({
  data: {
    email: 'elsa@prisma.io',
    name: 'Elsa Prisma',
  },
})
```

Show query results

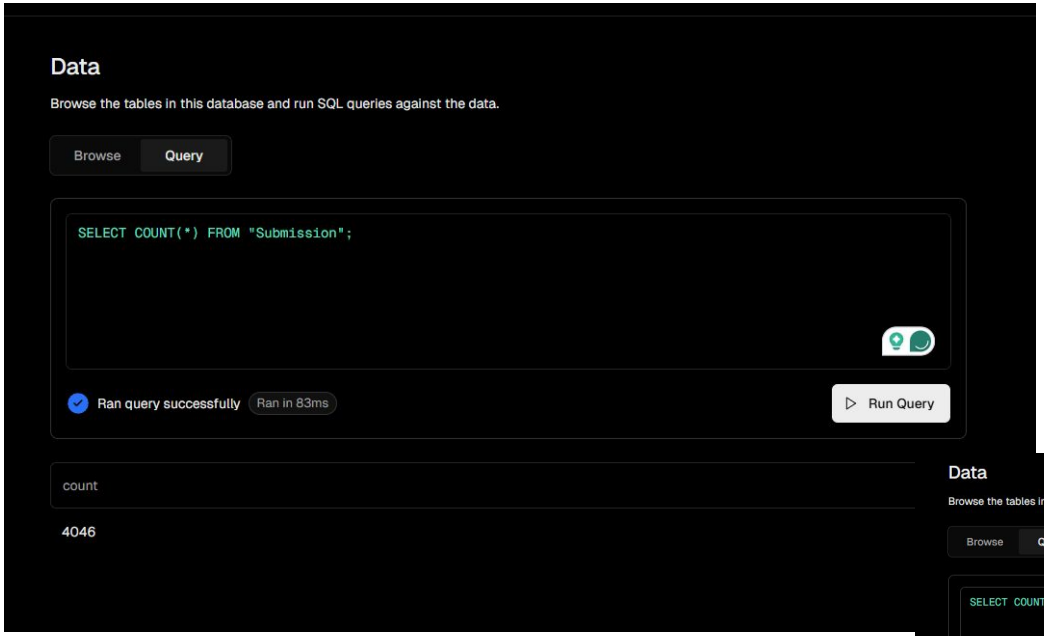
The following query returns all records where the `email` field contains `Prisma`, and sorts the result by the `title` field. The query skips the first 200 records and returns records 201 - 220.

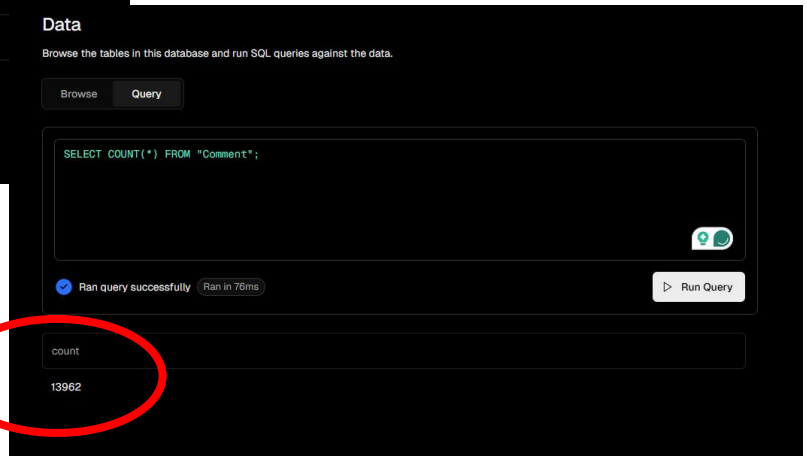
```
const results = await prisma.post.findMany({
  skip: 200,
  take: 20,
  where: {
    email: {
      contains: 'Prisma',
    },
  },
  orderBy: {
    title: 'desc',
  },
})
```

https://nextjs.org/

```
71                score: {
73                }
74              }
75            })
76          }}>
77            <ChevronUp className="h-4 w-4" />
78          </button>}
79          <div className="flex flex-col space-y-1">
80            {url ? (
81              <div className="flex items-center space-x-1">
82                <Link href={url} className="text-sm font-medium text-black hover:underline">
83                  {title}
84                </Link>
85                <span className="text-xs text-gray-500">
86                  ({new URL(url).hostname})
87                </span>
88              </div>
89            ) : (
90              <Link href={`/item/${id}`} className="text-sm font-medium text-black hover:underline">
91                {title}
92              </Link>
93            )}
94            <div className="flex items-center space-x-1 text-xs text-gray-500">
95              <span>{upvotes} upvotes</span>
96              <span>by</span>
97              <Link href={`/user/${author}`} className="hover:underline">
98                {author}
99              </Link>
100             <span>{time}</span>
101             <span>|</span>
102             <Link href={`/item/${id}`} className="hover:underline flex items-center">
103               <MessageSquare className="h-3 w-3 mr-1" />
104               {commentsCount} comments
105             </Link>
106             {session?.user?.email && <button className="text-xs text-gray-500 hover:text-[#ff6600]" onClick={async () => {
107               "use server"
108               await prisma.report.create({
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS 1 COMMENTS


Data
Browse the tables in this database and run SQL queries against the data.
Browse
Query
SELECT COUNT(*) FROM "Submission";
Ran query successfully
Ran in 83ms
Run Query
count
4046


Data
Browse the tables in this database and run SQL queries against the data.
Browse
Query
SELECT COUNT(*) FROM "Comment";
Ran query successfully
Ran in 76ms
Run Query
count
13962

DEMO

https://hndb.vercel.app/

test1@example.com password

Conclusion

ORMs are great

```
  time,
  commentsCount,
  rank
}: SubmissionProps) {
  const session = await auth()
  const ableToUpvote = session?.user?.email ? !await prisma.submissionUpvote.findFirst({
    where: {
      submissionId: id,
      user: {
        email: session.user.email
      }
    }
  }) : false

  return (
    <div className="flex items-start space-x-2 py-2">
      {rank && <span className="text-sm text-gray-500 w-4 text-right">{rank}.</span>}
```

Go look at code

https://github.com/jcbhmr/hndb