# 常规测试

|  |  |  |  |
| --- | --- | --- | --- |
| 编号 | 在name上建索引查询 | name上无index | 主键 |
| mariadb | 6.253732681274414 | 495.92053031921387 | 5.979753494262695 |
| mysql | 6.072957515716553 | 39.62403130531311 | 5.98876690864563 |
| **Postgresql** | **4.870428562164307** | **195.4096336364746** | **4.948631763458252** |

单位：s

# Json测试

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 编号 | 建索引 | | 无索引 | |
| Mariadb | 5.810502529144287 | | 1724.8685865402222 | |
| **Postgresql** | **Gin(在列上建索引 default )** | **Gin(在列上建索引 jsonb\_path\_ops)** | | **无索引** |
| **5.408046245574951** | **5.06257963180542** | | **690.3884344100952** |

单位：s

# 测试代码

postgresql json测试

|  |
| --- |
| #postgresql json测试  import psycopg2  import random  import time  conn = psycopg2.connect(database="postgres",user="postgres",password="...", host="192.168.40.129", port="5432")  cur = conn.cursor()  # cur.execute("explain analyze select \* from api where jdoc #> '{student}' @> '{\"name\":\"slohqiudvyktmcrxabezjwnpgf\"}'")  # rows = cur.fetchall()  # print(rows)  cur.execute("select jdoc ->'student'->'name' from api")  rows = cur.fetchall()  print(rows[0])  names = set()  for i in range(0, len(rows)):  names.add(rows[i][0])  # print(list(names)[0])  # cur.execute("drop table api")  # cur.execute("create table api(jdoc jsonb);")  # cur.execute("drop index hhh;")  # cur.execute("create index hhh on api using gin ((jdoc #> '{student}'));")  print(len(names))  # for i in range(270000):  # name = "".join(random.sample('zyxwvutsrqponmlkjihgfedcba',26))  # school = "".join(random.sample('zyxwvutsrqponmlkjihgfedcba',26))  # # print(name,school)  # while name in names:  # name = "".join(random.sample('zyxwvutsrqponmlkjihgfedcba', 26))  # names.add(name)  # # cur.execute("insert into api(jdoc) values('{\"student\": {\"name\": \"" +name + "\",\"school\":\"" + school + "\"}}');")  # cur.execute("insert into api2(jdoc) values ('{\"name\":\""+name+"\",\"school\":\""+school+"\"}');")  # print(i)  test\_list = list()  for i in range(1000):  test\_list.append(list(names)[random.randint(0, len(names))])  #  print('开始测试')  time\_start = time.time()  total\_time = 0  for i in range(10000):  name = test\_list[i%1000]  cur.execute("explain analyze select \* from api where jdoc @> '{\"student\":{\"name\":\"" + name + "\"}}';")  # rows = cur.fetchall()  # print(rows)  # total\_time = total\_time + float(rows[-1][0][16:-3])  # print(float(rows[-1][0][16:-3]))  time\_end = time.time()  print(time\_end-time\_start)  conn.commit()  conn.close()  print(cur) |

## mariadb json测试

|  |
| --- |
| Src/ |
| # mariadb json测试  import pymysql  import random  import time  db = pymysql.connect(host = "192.168.40.134", user="zx",passwd = "password",db = "mysql")  cur = db.cursor()  # cur.execute("drop table api")  # cur.execute("create table api(attr varchar(1024),CHECK (JSON\_VALID(attr)));")  # # cursor.execute("insert into api values('{\"name\": \"zx\", \"school\":\"nwpu\"}')")  # cur.execute("alter table api add name varchar(26) as (JSON\_VALUE(attr, \'$.name\'));")  cur.execute("select JSON\_VALUE(attr,\'$.name\') from api")  # cur.execute("create index namei on api(name);")  rows = cur.fetchall()  names = set()  for i in range(0, len(rows)):  names.add(rows[i][0])  # print(len(names))  # for i in range(270000):  # name = "".join(random.sample('zyxwvutsrqponmlkjihgfedcba',26))  # school = "".join(random.sample('zyxwvutsrqponmlkjihgfedcba',26))  # # print(name,school)  # print(i)  # if not name in names:  # names.add(name)  # cur.execute("insert into api(attr) values('{\"name\": \""+name+"\", \"school\":\"" + school + "\"}')")  # # print(len(names))  test\_list = list()  for i in range(1000):  test\_list.append(list(names)[random.randint(1,len(names))])  print("开始测试")  time\_start = time.time()  for i in range(10000):  name = test\_list[i%1000]  #print(name)  cur.execute("select \* from api where name = \"" +name+"\";")  time\_end = time.time()  print(time\_end - time\_start)  # print(tol\_time/1000)  # print(rows)  db.commit()  db.close() |

mysql 常规测试

|  |
| --- |
| #mysql 常规测试  import pymysql  import random  import time  db = pymysql.connect(host = "192.168.40.132", user="zx",passwd = "password",db = "mysql")  cur = db.cursor()  studentIDs = set()  cur.execute("select studentID from student;")  rows = cur.fetchall()  for i in range(len(rows)):  studentIDs.add(rows[i][0])  # for i in range(270000):  # name = "".join(random.sample('zyxwvutsrqponmlkjihgfedcba',20))  # school = "".join(random.sample('zyxwvutsrqponmlkjihgfedcba',20))  # studentID = random.randint(20000000, 99999999)  # while studentID in studentIDs:  # studentID = random.randint(200000000,999999999)  # studentIDs.add(studentID)  # cur.execute("insert into student(name,school,studentID) values(\""+name+"\","+"\""+school+"\","+str(studentID)+");")  # print(i)  test\_list = list()  for i in range(1000):  test\_list.append(list(studentIDs)[random.randint(0,len(studentIDs))])  # test\_list.append("".join(random.sample('zyxwvutsrqponmlkjihgfedcba',20)))  print("开始测试")  time\_start = time.time()  for i in range(10000):  # name = test\_list[i%1000]  # cur.execute("select \* from student where name" + "=\'" + name +"\';")  studentID = test\_list[i%1000]  cur.execute("select \* from student where studentID" + "=\'" + str(studentID) + "\';")  time\_end = time.time()  print(time\_end-time\_start)  print(cur)  db.commit()  db.close() |

postgresql 常规测试

|  |
| --- |
| #postgresql 常规测试  import psycopg2  import random  import time  conn = psycopg2.connect(database="postgres",user="postgres",password="...", host="192.168.40.129", port="5432")  cur = conn.cursor()  studentIDs = set()  cur.execute("select studentID from student;")  rows = cur.fetchall()  for i in range(0, len(rows)):  studentIDs.add(rows[i][0])  # print(list(studentIDs)[0])  # for i in range(270000):  # name = "".join(random.sample('zyxwvutsrqponmlkjihgfedcba',20))  # school = "".join(random.sample('zyxwvutsrqponmlkjihgfedcba',20))  # studentID = random.randint(20000000, 99999999)  # while studentID in studentIDs:  # studentID = random.randint(200000000,999999999)  # studentIDs.add(studentID)  # cur.execute("insert into student (name, school, studentID) values (\'"+name+"\',"+"\'"+school+"\',"+str(studentID)+");")  # print(i)  test\_list = list()  for i in range(1000):  # test\_list.append(list(studentIDs)[random.randint(0,len(studentIDs))])  # name  test\_list.append("".join(random.sample('zyxwvutsrqponmlkjihgfedcba',20)))  print("开始测试")  time\_start = time.time()  for i in range(10000):  name = test\_list[i%1000]  cur.execute("select \* from student where name" + "=\'" + name + "\';")  # studentID = test\_list[i%1000]  # cur.execute("select \* from student where studentID="+ str(studentID)+";")  time\_end = time.time()  print(time\_end-time\_start)  conn.commit()  conn.close() |

mariadb 常规测试

|  |
| --- |
| #mariadb 常规测试  import pymysql  import random  import time  db = pymysql.connect(host = "192.168.40.134", user="zx",passwd = "password",db = "mysql")  cur = db.cursor()  studentIDs = set()  cur.execute("select studentID from student;")  rows = cur.fetchall()  for i in range(len(rows)):  studentIDs.add(rows[i][0])  # for i in range(270000):  # name = "".join(random.sample('zyxwvutsrqponmlkjihgfedcba',20))  # school = "".join(random.sample('zyxwvutsrqponmlkjihgfedcba',20))  # studentID = random.randint(20000000, 99999999)  # while studentID in studentIDs:  # studentID = random.randint(200000000,999999999)  # studentIDs.add(studentID)  # cur.execute("insert into student(name,school,studentID) values(\""+name+"\","+"\""+school+"\","+str(studentID)+");")  # print(i)  test\_list = list()  for i in range(1000):  test\_list.append(list(studentIDs)[random.randint(0,len(studentIDs))])  # name  # test\_list.append("".join(random.sample('zyxwvutsrqponmlkjihgfedcba',20)))  print("开始测试")  # time\_start = time.time()  for i in range(10000):  # name = test\_list[i%1000]  # cur.execute("select \* from student where name" + "=\'" + name + "\';")  studentID = test\_list[i%1000]  cur.execute("select \* from student where studentID" + "=\'" + str(studentID) +"\';")  time\_end = time.time()  print(time\_end-time\_start)  db.commit()  db.close() |