



دانشکده مهندسی کامپیوتر

گروه مهندسی نرم افزار

عنوان پروژه:

آزمون کارایی پایگاه داده

Database Performance Testing

پیش نویس اول پروژه کلاسی شماره ۳ درس مهندسی نرم افزار پیشرفته

دانشجویان:

مرتضی ذاکری - محسن امیریان - حمیدرضا احمدی

استاد:

دکتر سعید پارسا

اردیبهشت ماه ۱۳۹۶

فهرست مطالب

۱.....	مقدمه	۱
۱.....	آزمون کارایی پایگاه داده	۱-۱
۲.....	ابزارهای آزمون پایگاه داده	۲-۱
۳.....	نرم افزار تحت آزمون	۲
۷.....	تولید خودکار داده‌های آزمون	۳
۱۱.....	استفاده از ابزار APACHE JMETER برای آزمون بار پایگاه داده	۴
۱۱.....	آماده سازی JMeter برای ارتباط با Microsoft SQL Server	۱-۱-۴
۱۲.....	طرح آزمون	۲-۴
۱۴.....	سناریوی آزمون	۱-۲-۴
۱۵.....	افزودن کاربران	۲-۲-۴
۱۷.....	افزودن درخواست پرس و جوی پایگاه داده	۳-۲-۴
۲۲.....	افزودن عنصر Listener و مشاهده نتایج آزمون	۴-۲-۴
۲۴.....	اجرای آزمون و تحلیل نتایج	۵
۲۴.....	آزمایش اول	۱-۵
۲۵.....	آزمایش دوم	۲-۵
۲۷.....	مراجع	۶

فهرست شکل ها

- شکل ۱-۲ نمودار ER پایگاه داده سامانه مدیریت برگزاری مسابقات شنا (بخش مدیریت کاربران)..... ۴
- شکل ۲-۲ نمودار ER پایگاه داده سامانه مدیریت برگزاری مسابقات شنا (بخش برگزاری مسابقه و ثبت نتایج)..... ۵
- شکل ۳-۲ نمودار ER پایگاه داده سامانه مدیریت برگزاری مسابقات شنا (بخش مدیریت تیمها و شناگران)..... ۶
- شکل ۱-۳ ابزارهای مختلف بسته نرم افزاری SQL TOOLBELT از شرکت RED-GATE..... ۸
- شکل ۲-۳ ورود به ابزار تولید داده RED-GATE..... ۹
- شکل ۳-۳ تنظیمات نحوه تولید داده ها..... ۱۰
- شکل ۴-۳ داده‌های تصادفی تولید شده برای جدول شناگران..... ۱۰
- شکل ۱-۴ درایور اتصال با پایگاه داده از طریق استاندارد JDBC..... ۱۲
- شکل ۲-۴ نحوه اضافه کردن درایور JDBC به کتابخانه JMETER..... ۱۲
- شکل ۳-۴ اضافه کردن عنصر به طرح آزمون در JMETER..... ۱۳
- شکل ۴-۴ افزودن کاربران به طرح آزمون..... ۱۶
- شکل ۵-۴ گزینه های THREAD GROUP..... ۱۶
- شکل ۶-۴ تنظیمات کنترل پنل کاربران..... ۱۷
- شکل ۷-۴ اضافه کردن یک اتصال JDBC برای ارسال پرس و جوهای پایگاه داده..... ۱۸
- شکل ۸-۴ تنظیمات اتصال شامل رشته اتصال، درایور، شناسه و گذرواژه مدیر پایگاه..... ۱۹
- شکل ۹-۴ نحوه ساخت پرس و جوی مربوط به نشان دادن نتایج انفرادی در محیط SQL SERVER MANAGEMENT STUDIO..... ۲۰
- شکل ۱۰-۴ اضافه کردن یک درخواست پرس و جو به طرح آزمون JMETER..... ۲۱
- شکل ۱۱-۴ نحوه ساخت پرس و جوی مربوط به نشان دادن نتایج تیمی در محیط SQL SERVER MANAGEMENT STUDIO..... ۲۱
- شکل ۱۲-۴ اضافه کردن درخواست پرس و جوی دوم به طرح آزمون JMETER..... ۲۲
- شکل ۱۳-۴ افزودن یک عنصر LISTENER برای مشاهده نتایج آزمون..... ۲۳
- شکل ۱۴-۴ مشاهده نتایج به صورت جدولی..... ۲۳
- شکل ۱-۵ گزارش خلاصه جزئیات آزمون برگزار شده..... ۲۴
- شکل ۲-۵ جزئیات کامل درخواست های فرستاده شده و پاسخ های دریافتی در طی آزمون..... ۲۵
- شکل ۳-۵ وضعیت پردازنده مرکزی در حین انجام آزمایش دوم..... ۲۶
- شکل ۴-۵ خلاصه نتایج آزمایش دوم..... ۲۶

فهرست جدول ها

- جدول ۱-۱ ابزارهای مورد استفاده در آزمون های مختلف پایگاه داده ها ۲
- جدول ۱-۲ مشخصات سخت افزاری و نرم افزاری محیط آزمون ۷
- جدول ۱-۴ رشته اتصال و کلاس درایور برای ارتباط JMETER با پایگاه داده از طریق JDBC برای برخی پایگاه داده های شناخته شده ۱۹

۱ مقدمه

۱-۱ آزمون کارایی پایگاه داده

آزمون پایگاه داده شامل بررسی صحت^۱ و جامعیت^۲ داده‌ها، بررسی کارایی^۳ پایگاه داده و آزمودن رویه‌ها^۴، محرک‌ها^۵ و توابع موجود در یک پایگاه داده است. در این میان بررسی کارایی از اهمیت به‌سزایی برخوردار است. معیارهای کارایی معمول عبارت‌اند از زمان پاسخ، گذردهی و بهره‌وری که باید به‌صورت مقتضی در شرایط گوناگون سنجیده شوند و سیستم به‌نحوی طراحی شود که این معیارها مقادیر قابل قبولی داشته باشند.

یک برنامه کاربردی با زمان پاسخ زیاد و به تبع آن کارایی ضعیف می‌تواند عامل ایجاد بسیاری از مشکلات مشاهده شده از سوی کاربران آن برنامه باشد. آزمون کارایی سیستم‌ها معمولاً به دو شاخه آزمون بار^۶ و آزمون فشار^۷ تقسیم می‌شود. این شاخه‌ها برای آزمون کارایی پایگاه داده نیز به‌کار می‌روند.

هدف اولیه آزمون بار پایگاه داده بررسی این است که افزایش تدریجی تعداد تراکنش‌ها چه تأثیری بر مقادیر معیارهای کارایی می‌گذارد. در آزمون بار جنبه‌های زیر بایستی بررسی شوند:

- زمان پاسخ برای اجرای تراکنش‌ها توسط تعدادی کاربر از راه دور.
- درصد استفاده از منابع مختلف جهت پاسخ‌گویی به تعدادی تراکنش.
- تعداد تراکنش‌های قابل اجرا به صورت همزمان توسط کاربران.

آزمون فشار با هدف تشخیص نقطه شکست^۸ سیستم صورت می‌پذیرد. در این آزمون برنامه تحت بار کاری قرار می‌گیرد که سرانجام آن را از کار می‌اندازد. نقطه‌ای که برنامه تحت آزمون در آن کاملاً از کار افتاده محسوب می‌شود، نقطه شکست نامیده می‌شود. بر مبنای نتایج حاصله بایستی

Validity^۱

Integrity^۲

Performance^۳

Procedures^۴

Triggers^۵

Load Testing^۶

Stress Testing^۷

breakpoint^۸

نسبت به رفع مشکلات موجود اقدام نمود. در پایگاه داده معمولاً این مشکلات با بهینه سازی پرس‌وجوها، شاخص گذاری‌های مناسب، انواع داده‌ای مناسب و غیره مرتفع می‌شوند.

۲-۱ ابزارهای آزمون پایگاه داده

ابزارهای گوناگونی توسط شرکت‌های مختلف برای تولید داده‌های آزمون، مدیریت داده‌های آزمون و انجام عملیات آزمون نظیر آزمون بار و فشار برای پایگاه داده فراهم شده اند. یکی از ابزارهای ساده و در عین پر کاربرد در این زمینه ابزار JMeter از بنیاد نرم افزارهای آزاد آپاچی است که برای آزمون بار برنامه‌های تحت وب و نیز پایگاه داده قابل استفاده می‌باشد. در این گزارش به بررسی چگونگی استفاده از ابزار JMeter به منظور انجام آزمون کارایی پایگاه داده می‌پردازیم. در جدول ۱ تعدادی دیگر از ابزارهای مطرح در انواع آزمون‌های پایگاه داده‌ها نشان داده شده است.

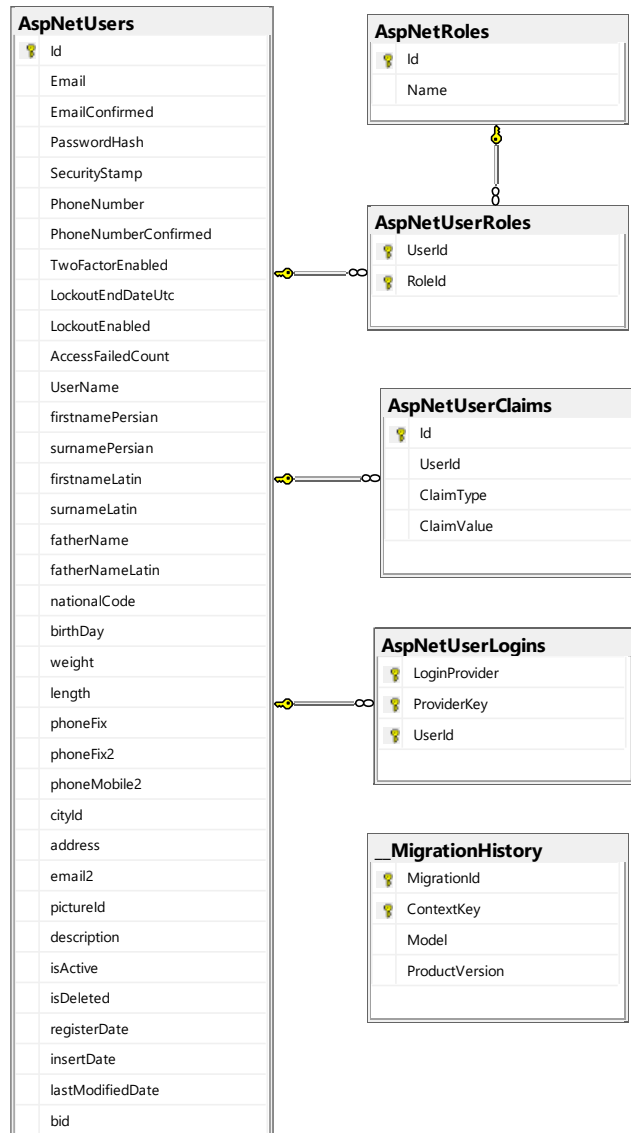
جدول ۱-۱ ابزارهای مورد استفاده در آزمون‌های مختلف پایگاه داده‌ها

ردیف	نوع آزمون و توضیحات	ابزارهای نمونه
۱	ابزارهای آزمون بار	<ul style="list-style-type: none"> • Web Performance • Rad View • Mercury • Apache JMeter <p>این ابزارهای برای فرستادن بارکاری و داده‌ای زیاد به پایگاه داده برنامه به کار می‌روند، که کمک می‌کنند تا بتوان مشخص نمود آیا مشخصات سخت افزاری و نرم افزاری سیستم در شرایط کاری سنگین پاسخ‌گوی نیازهای برنامه کاربردی مورد نظر هستند.</p>
۲	ابزارهای امنیت داده	<ul style="list-style-type: none"> • IBM Optim Data Privacy <p>این ابزارها به منظور پیاده‌سازی موافقت‌نامه‌ها و استانداردهای موجود برای تضمین امنیت داده‌ها به کار می‌روند.</p>
۳	ابزارهای تولید داده‌های آزمون	<ul style="list-style-type: none"> • Data Factory • DTM Data Generator • Turbo Data <p>آزمون‌کنندگان این ابزارها را برای تولید داده‌های آزمایشی مورد استفاده در آزمون به کار می‌برند. این مورد زمانی نیاز است که شما به مقدار زیادی نمونه از داده‌ها نیاز دارید تا آزمون مورد نظر</p>

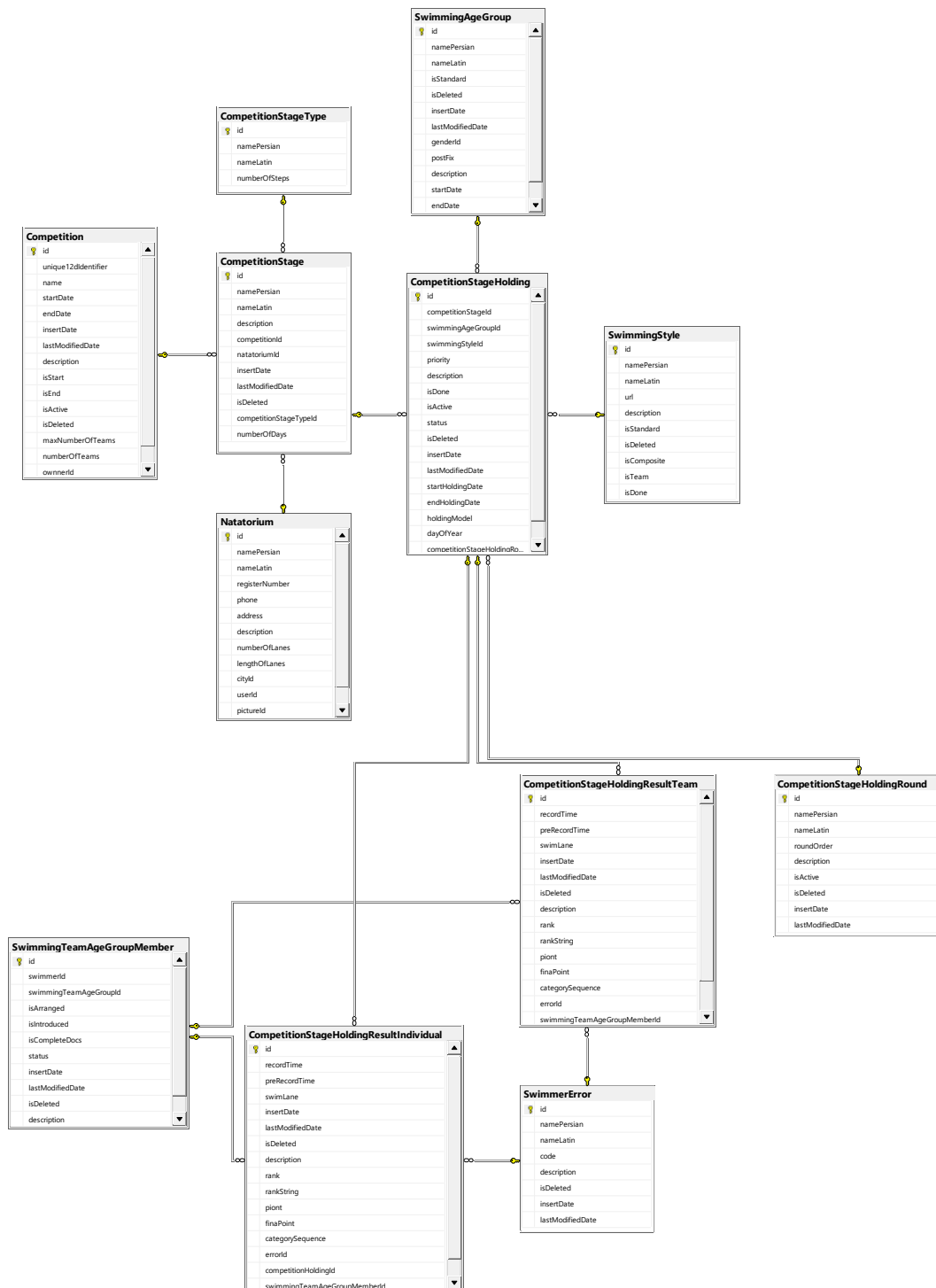
• Red-Gate SQL Data Generator	خود را روی پایگاه داده انجام دهید. تولید داده‌های آزمون در آزمون‌های کارایی و فشار بسیار استفاده می‌شود.	
• IBM Optim Test Data Management	ابزارهای مدیریت داده‌های آزمون	۴
	این ابزارها به منظور کنترل نسخه داده‌های آزمون استفاده می‌شوند. در مواردی که نیاز است تا آزمون‌گر نتایج مورد انتظار را مشخص کند و سپس آن‌ها را با خروجی حاصل از آزمون مقایسه کند، نیاز است تا داده‌های آزمون و خروجی‌ها را در نسخه‌های متعددی نگهداری کنیم.	
• SQLUnit • TSQLUnit • DBFit • DBUnit	ابزارهای انجام آزمون یکه	۵
	این ابزارها برای انجام آزمون یکه ^۹ بر روی قسمت‌های مختلف پایگاه داده به کار می‌روند.	

۲ نرم افزار تحت آزمون

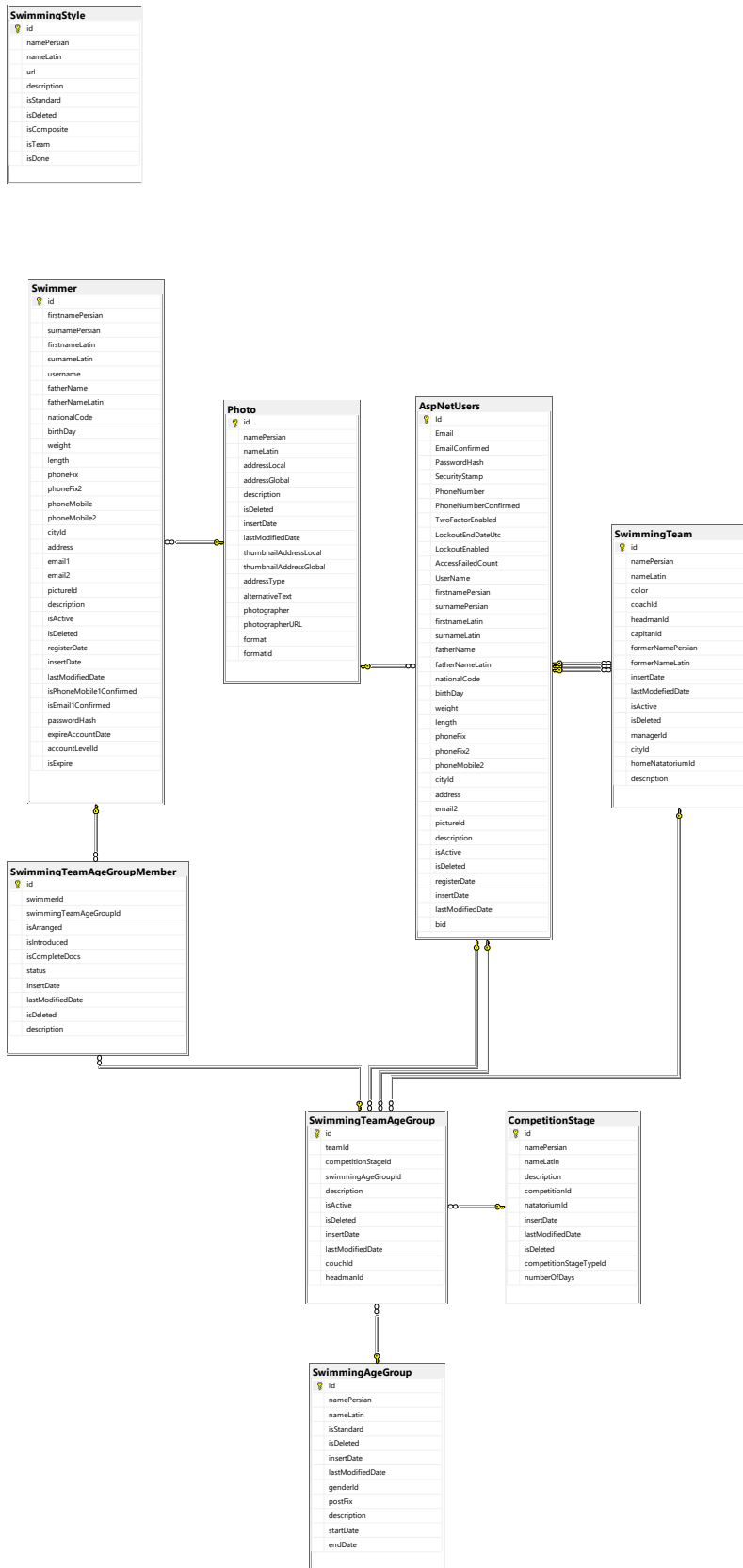
در این بخش مشخصات نرم‌افزار تحت آزمون و نیز سیستمی که آزمون بر روی آن انجام شده است را بیان می‌کنیم. به عنوان مطالعه موردی در این گزارش پایگاه داده پروژه مدیریت برگزاری مسابقات شنا که در پروژه کلاسی شماره ۲ معرفی شد را مورد آزمون کارایی قرار می‌دهیم. شمای پایگاه داده در نرم‌افزار مدیریت پایگاه داده Microsoft SQL Server 2016 تعریف شده است و شامل ۲۸ جدول است. شکل ۲-۱، شکل ۲-۲ و شکل ۲-۳ نمودار ER بخش‌های اصلی شمای پایگاه مورد آزمون را در قالب نمودار ER نشان می‌دهند. همچنین جدول ۱-۱ و جدول ۱-۲ مشخصات سخت‌افزاری و نرم‌افزاری سیستمی که آزمون روی آن انجام شده است را نشان می‌دهد.



شکل ۱-۲ نمودار ER پایگاه داده سامانه مدیریت برگزاری مسابقات شنا (بخش مدیریت کاربران)



شکل ۲-۲ نمودار ER پایگاه داده سامانه مدیریت برگزاری مسابقات شنا (بخش برگزاری مسابقه و ثبت نتایج)



شکل ۲-۳ نمودار ER پایگاه داده سامانه مدیریت برگزاری مسابقات شنا (بخش مدیریت تیمها و شناگران)

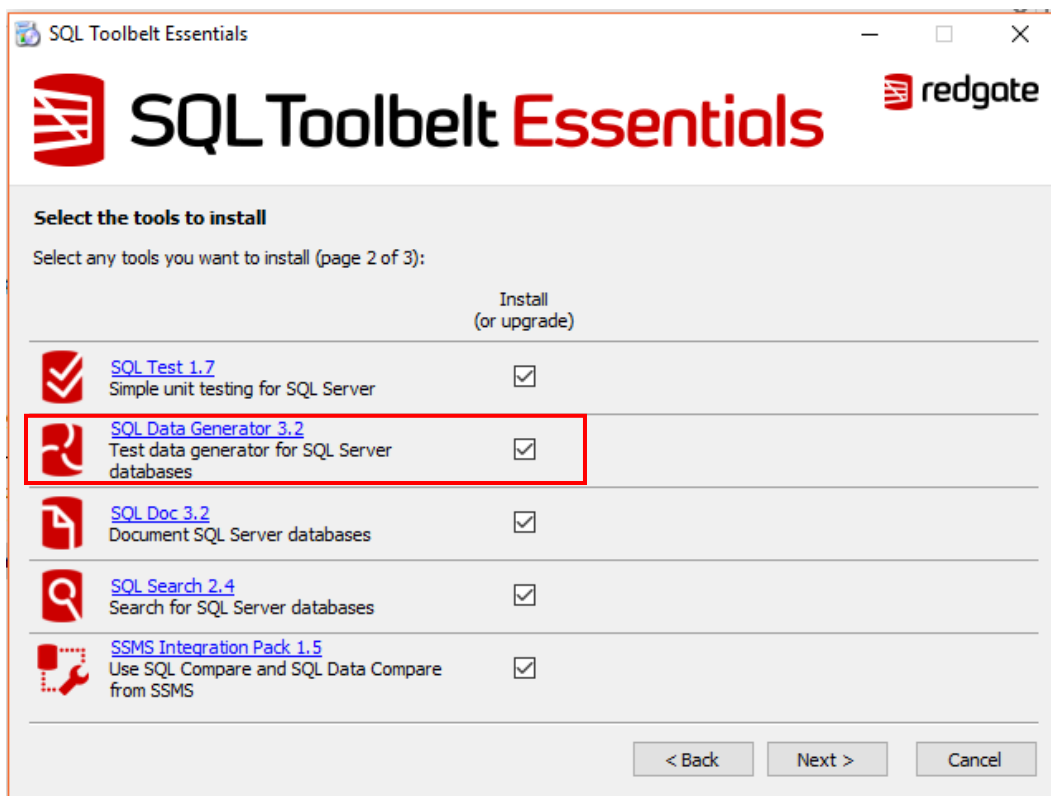
جدول ۱-۲ مشخصات سخت‌افزاری و نرم‌افزاری محیط آزمون

Intel®Core™ i7-6700HQ CPU @ 2.60GHz	پردازنده مرکزی
16.0 GB DDR3 RAM	حافظه اصلی (RAM)
ADATA SSD SP550NS38 120GB	حافظه جانبی
Microsoft® Windows 10 Enterprise 2016	سیستم عامل
Microsoft® SQL Server 2016	موتور پایگاه داده

۳ تولید خودکار داده‌های آزمون

همان‌طور که در بخش اول بیان شد، آزمون کارایی نیازمند حجم انبوهی از داده‌های آزمون است. از آنجایی که آزمون کارایی پایگاه داده‌ها معمولاً در محیط عملیاتی صورت می‌گیرد؛ داده‌هایی در جداول پایگاه داده وجود دارند که چنانچه کافی باشند می‌توان آنها را مورد استفاده قرار داد. با توجه به اینکه ما در مرحله توسعه هستیم و داده‌ای در جداول وجود ندارد، بهتر است از ابزارهایی برای تولید داده‌های فرضی جهت پر کردن جدول‌های مختلف پایگاه داده استفاده شود.

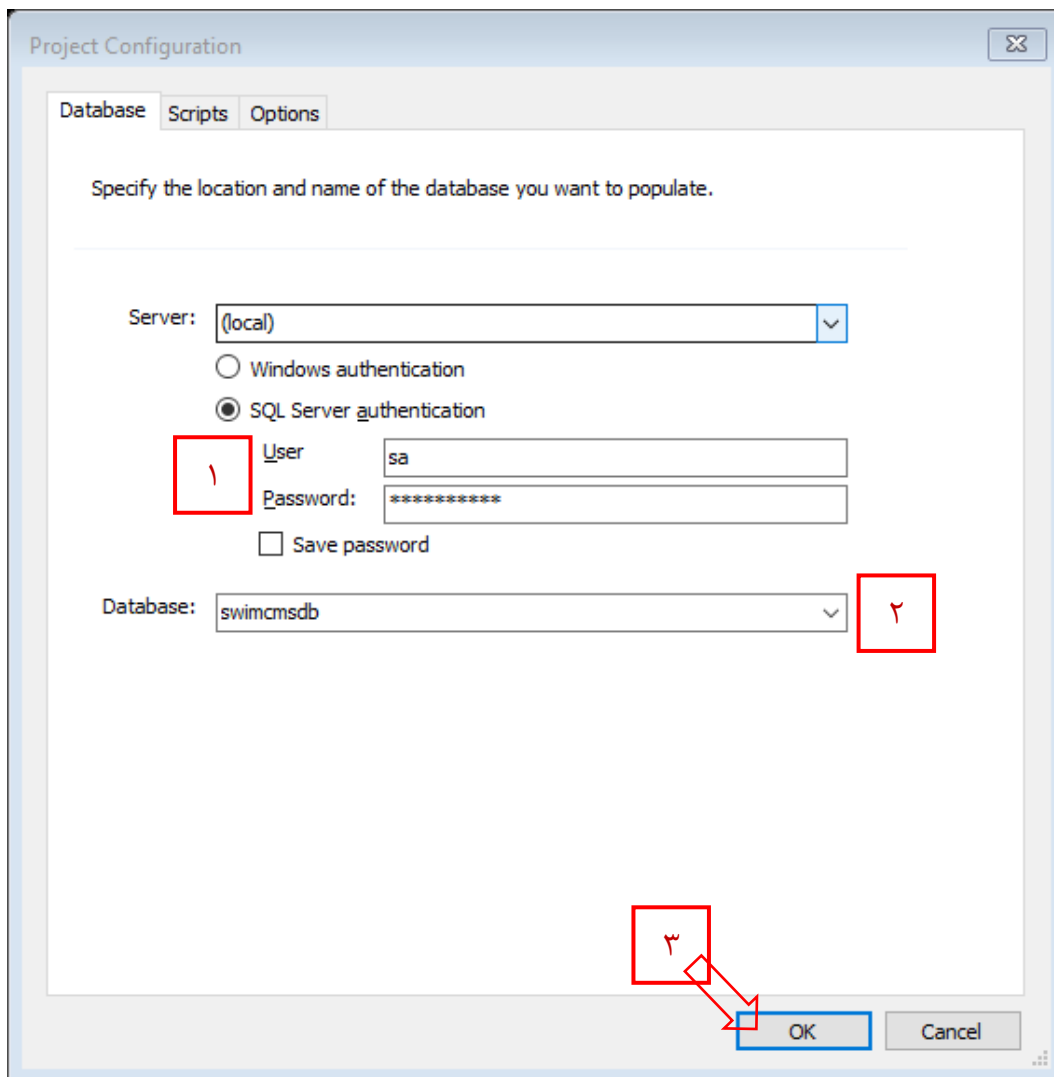
ابزارهای مختلفی برای این منظور در جدول ۱-۱ معرفی شدند. در این قسمت ما از نرم‌افزار SQL Toolbelt محصول شرکت RedGate استفاده می‌کنیم. SQL Toolbelt ابزاری به نام SQL Data Generator فراهم می‌کند که برای تولید داده‌های آزمایشی استفاده می‌شود.



شکل ۱-۳ ابزارهای مختلف بسته نرم‌افزاری SQL Toolbelt از شرکت Red-Gate

پس از نصب و پیکربندی نرم‌افزار Microsoft SQL Server 2016 نرم‌افزار RedGate SQL Toolbelt را مطابق شکل ۱-۳ نصب می‌کنیم. دقت شود که بایستی علامت نصب SQL Data Generator حتما خورده باشد.

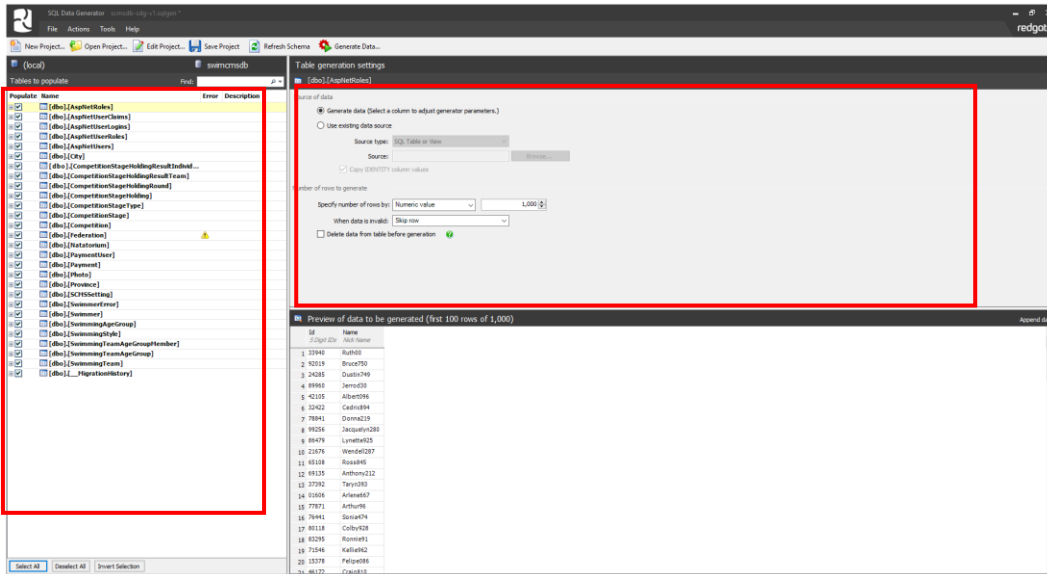
پس از نصب با ساخت یک حساب کاربری می‌توان به صورت آزمایشی به مدت ۱۴ روز از نرم‌افزار استفاده کرد. برنامه SQL Data Generator 3 را باز می‌کنیم. با وارد کردن نام کاربری و رمز عبور مدیر پایگاه داده به موتور پایگاه داده متصل می‌شویم و سپس پایگاه داده مربوطه را انتخاب می‌کنیم. این مراحل به ترتیب شماره‌گذاری و در شکل ۲-۳ نمایش داده شده است.



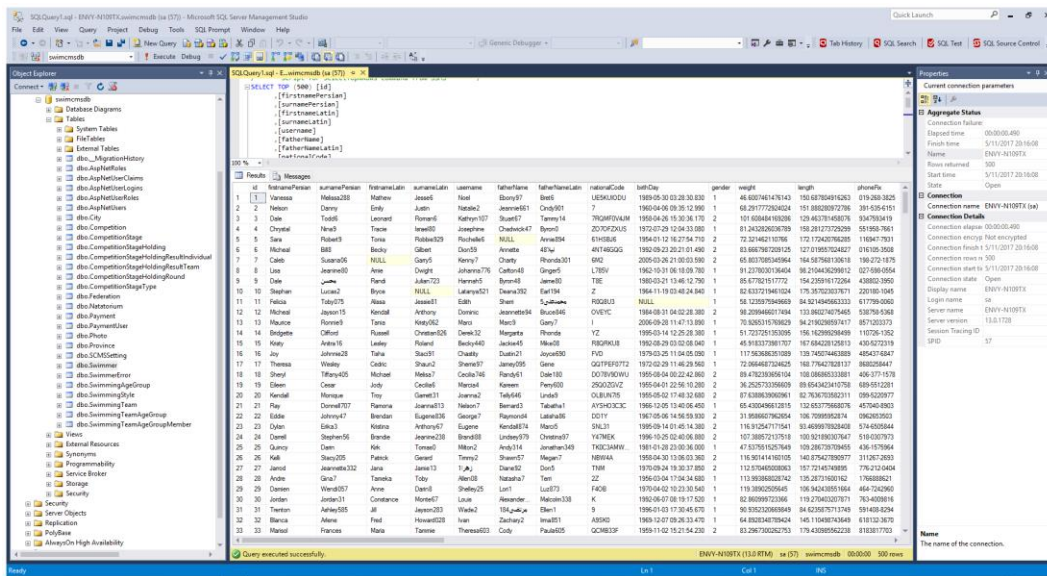
شکل ۳-۲ ورود به ابزار تولید داده Red-Gate

بعد از ورود به برنامه یک پروژه ایجاد می‌شود. در قسمت سمت چپ پنجره اصلی برنامه اسامی کلیه جدول‌ها پایگاه داده انتخابی مشاهده می‌شود. جدول‌هایی را که می‌خواهیم برای آن‌ها رکورد تولید شود، انتخاب می‌کنیم. در قسمت سمت راست پنجره اصلی می‌توان تعداد رکوردهایی تولید برای هر جدول را به عدد وارد کرد. همچنین می‌توان این تعداد را بر حسب زمان اجرای تولید رکورد مطرح کرد. یعنی به عنوان مثال به مدت ۱۰ ثانیه داده تولید نما و در پایگاه داده ذخیره کن. این تنظیمات در شکل ۳-۳ نشان داده شده است.

آزمون کارایی پایگاه داده‌ها



شکل ۳-۳ تنظیمات نحوه تولید داده‌ها



شکل ۳-۴ داده‌های تصادفی تولید شده برای جدول شناسگران

علاوه بر تنظیمات گفته شده در بالا، برای هر فیلد یا ستون موجود در هر یک از جدول‌ها می‌توان محدوده مقادیر تولید شده را مشخص نمود. به صورت پیش فرض یک درصد مقادیر هر ستون در کل مقادیر تولید شده، مقدار null خواهند داشت که این مورد نیز قابل تغییر است. پس از اعمال همه تنظیمات و ذخیره پروژه گزینه Generate Data از نوار ابزار می‌زینم. بخشی از نتیجه حاصل شده برای جدول شناسگر در شکل ۳-۴ آمده است.

۴ استفاده از ابزار Apache JMeter برای آزمون بار پایگاه داده

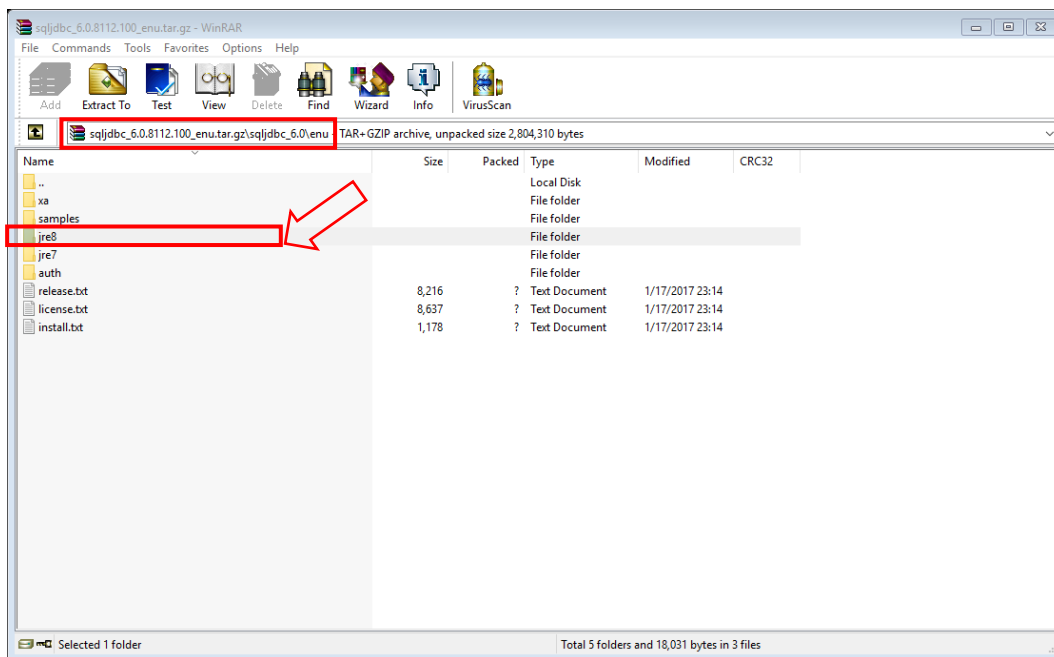
در بخش قبلی داده‌های آزمون مورد نظر خود را تولید کرده و پایگاه داده را با این داده‌ها پر کردیم. در این بخش نحوه استفاده از JMeter در آزمون کارایی (به طور دقیق‌تر آزمون بار) پایگاه داده را با اجرای یک سناریوی آزمون روی سامانه مدیریت برگزاری مسابقات شنا که در بخش دو معرفی شد، شرح می‌دهیم. می‌خواهیم عملکرد پایگاه داده طراحی شده در بارهای کاری مختلف را مورد آزمایش قرار دهیم.

۱-۱-۴ آماده سازی JMeter برای ارتباط با Microsoft SQL Server

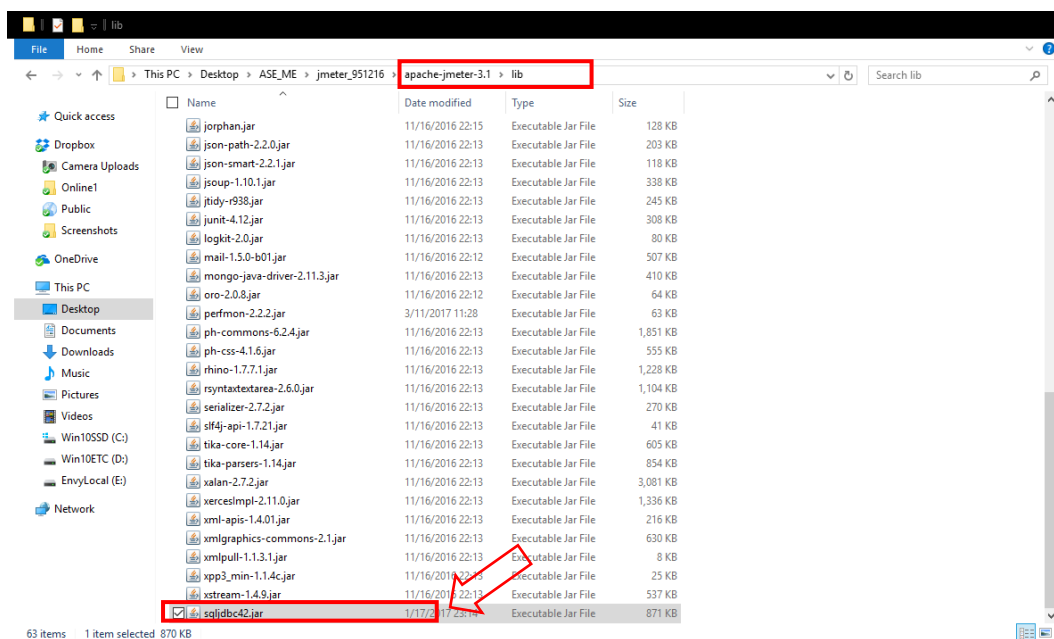
JMeter ابزاری است که به زبان جاوا نوشته شده است. به همین دلیل از طریق واسط JDBC با پایگاه داده‌ها ارتباط برقرار می‌کند. JDBC برای پایگاه داده‌های مختلفی توسعه داده شده است. واسط JDBC مختص Microsoft SQL Server 2016 از نشانی زیر قابل دریافت است:

- <https://www.microsoft.com/en-us/download/details.aspx?id=11774>

فایل فشرده حاوی sqljdbc را دریافت کرده و نسخه متناسب با جاوا سیستم خود را در دایرکتوری lib ابزار JMeter کپی می‌نماییم. این مراحل در شکل ۱-۴ و شکل ۲-۴ به ترتیب نشان داده شده است.



شکل ۴-۱ درایور اتصال با پایگاه داده از طریق استاندارد JDBC



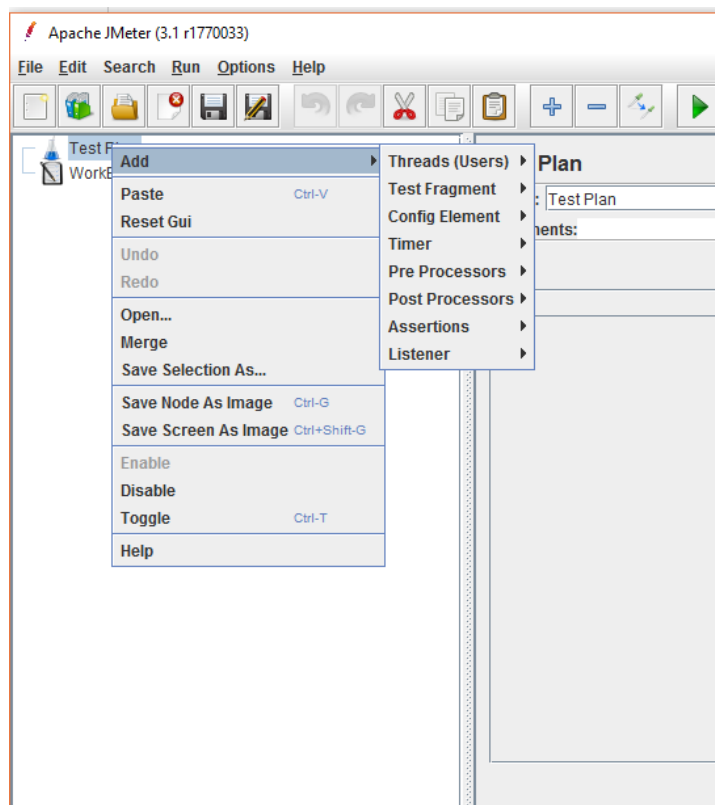
شکل ۴-۲ نحوه اضافه کردن درایور JDBC به کتابخانه JMeter

۲-۴ طرح آزمون

پس از آماده سازی JMeter، مرحله بعد ایجاد یک طرح آزمون^{۱۰} در JMeter است. طرح آزمون یک سری گام را توصیف می‌کند که JMeter هنگام اجرای آزمون کارایی آن‌ها را می‌پیماید. در

واقع JMeter این گام‌ها را متوالی طی می‌کند. یک طرح آزمون کامل متشکل از یک یا تعدادی بیش‌تر Thread Groups، Logic Controllers، Sample Generating Controllers، Listener، Timers، Assertions و Configuration Elements است که هر قسمت را در محل خود توضیح خواهیم داد. به طور کلی هریک از قسمت‌های نام برده شامل تعدادی عنصر هستند. اصولاً شروع کار با JMeter برای انجام هرگونه آزمون کارایی با ساخت طرح آزمون و سپس اضافه کردن عناصر به آن همراه است.

اضافه کردن یک عنصر به طرح آزمون در حالت گرافیکی برنامه JMeter به راحتی و با کلیک راست بر روی گزینه Test Plan در بالای سمت چپ پنجره اصلی برنامه انجام می‌شود. در شکل ۳-۴ نحوه انجام این کار نشان داده شده است.



شکل ۳-۴ اضافه کردن عنصر به طرح آزمون در JMeter

۱-۲-۴ سناریوی آزمون

سناریو کلی آزمون باری که می‌خواهیم بر روی پایگاه داده سامانه برگزاری مسابقات شنا اجرا کنیم بدین شرح است. می‌خواهیم تعداد ۱۰۰ کاربر فرضی ایجاد کنیم که تعداد دو درخواست^{۱۱} SQL را به سرویس‌دهنده پایگاه داده سامانه می‌فرستند. همچنین فرض می‌کنیم هر کاربر درخواست خود را ۵ مرتبه تکرار می‌کند. بنابراین تعداد کل درخواست‌های ارسالی به سمت سرویس‌دهنده پایگاه داده برابر خواهد بود با:

- Total Number of DB Request = 100 Users X 2 SQL Commands X 5 Repeat Times = 1000 JDBC Request

دقت شود که در ما در این مرحله در حال طراحی یک طرح آزمون به صورت آزمایشی هستیم و برای اجرای واقعی این آزمون بایستی تعداد کاربران، فرمان‌های SQL و نیز تعداد تکرار آن‌ها افزایش یابد. در نتیجه با افزایش این مقادیر و به طبع آن افزایش تعداد درخواست‌ها به سرویس‌دهنده پایگاه داده، منابع سیستم را پایش کرده و معیارهای کارایی را محاسبه می‌کنیم. در صورت مناسب نبودن هریک از آن‌ها باید منشا مشکل را پیدا کرده و اقدامات لازم را جهت برطرف سازی آن انجام دهیم. همچنین می‌توانیم سخت‌افزار لازم برای سرویس‌دهنده را تعیین کنیم.

اما انگیزه ما از مطرح کردن سناریوی آزمون به شکل فوق چیست؟ سامانه برگزاری مسابقات شنا در حالت عادی بار کاری زیادی را متحمل نمی‌شود، اما در زمان برگزاری مسابقه شنا امکان افزایش بار کاری سامانه وجود دارد. یکی از قابلیت‌هایی که در این سامانه پیش‌بینی شده است، امکان مشاهده رکوردهای شناگران و نتایج رده‌بندی آن‌ها بلافاصله پس از برگزاری مسابقه و از طریق اینترنت به وسیله کاربران و شناگران حاضر در مسابقه است.

مسابقات شنا از جهت برگزاری به دو صورت انفرادی و تیمی تقسیم می‌شوند که نتایج هریک از آن‌ها در جداول جداگانه‌ای ثبت می‌شود. در نتیجه انتظار می‌رود کاربران بعد از برگزاری هر دور مسابقه جهت مشاهده نتایج آن دور به سامانه تحت وب مراجعه کنند. اگر فرض کنیم که تعداد ۱۰۰ کاربر و شناگر برای مشاهده نتایج نهایی مسابقه، به سامانه مراجعه کرده و هریک دو گزینه مشاهده نتایج تیمی و مشاهده نتایج انفرادی (معادل دو پرس‌وجوی مجزا روی پایگاه داده برنامه) را بزنند و این کار را به علتی نظیر نشان دادن نتایج به دیگران، ذخیره یا چاپ آن،

^{۱۱} Request

اختلالات شبکه نظیر قطعی اینترنت و غیره، ۵ مرتبه انجام دهند؛ آنگاه سناریو فوق در عمل شکل می‌گیرد.

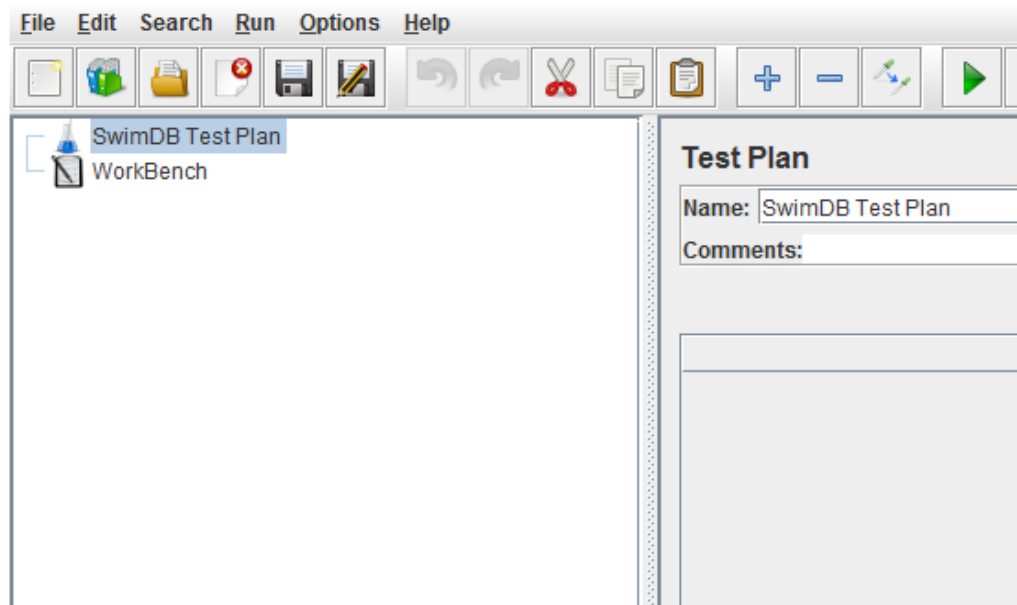
پس وقوع چنین سناریویی و به تبع آن سناریوهای مشابه در دنیای واقعی و استفاده عملی سامانه خیلی محتمل است و لذا بررسی رفتار سامانه در شرایط نام برده بسیار حایز اهمیت می‌باشد.

۲-۲-۴ افزودن کاربران

اولین مرحله در آغاز ایجاد یک طرح آزمون افزودن کاربران است. در واقع با این کار، طرح آزمون ساخته شده و اولین عنصر نیز به آن افزوده شده است. برای افزودن کاربران لازم است یک عنصر Thread Group به طرح آزمون اضافه شود. Thread Group به JMeter می‌گوید که چه تعداد کاربر فرضی می‌خواهید ایجاد کنید و همچنین اینکه هر کاربر فرضی چند درخواست و هر درخواست را چند بار باید ارسال کنند.

برای ایجاد Thread Group کافی است طرح آزمون مورد نظر را انتخاب کرده، بر روی آن کلیک راست کنیم و از قسمت Add گزینه ThreadGroup را انتخاب کنیم (شکل ۴-۴). پس از این کار می‌توان عنصر ThreadGroup را در زیر طرح آزمون دید. توجه داشته باشید که لیست باید در حالت expand باشد. سپس نیاز است جزئیات پیش فرض را تغییر دهیم. با انتخاب ThreadGroup گزینه‌های شکل ۴-۵ را مشاهده می‌کنید.

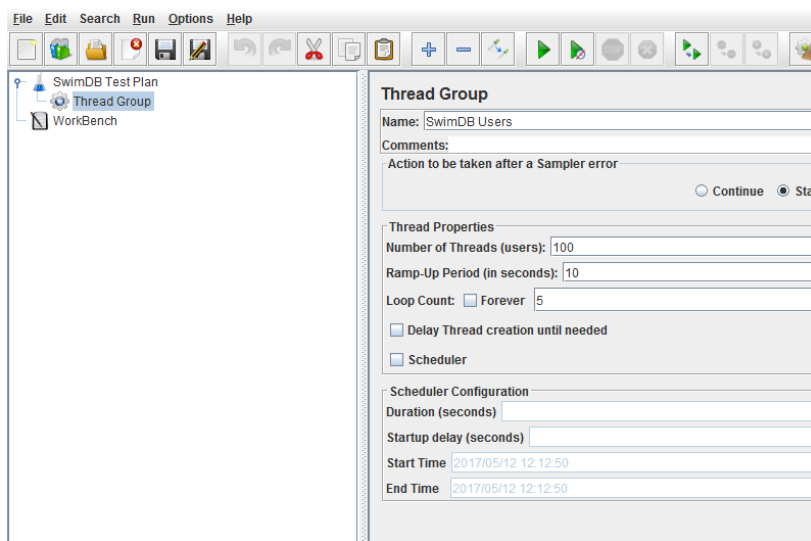
سپس تعداد کاربران را به ۱۰۰ کاربر افزایش می‌دهیم. در فیلد بعدی، Ramp-Up period را با ۱۰ ثانیه مقداردهی می‌کنیم. این فیلد زمان لازم برای شروع به کار تمام کاربران را مشخص می‌کند. یعنی زمانی که ما این مقدار را ۱۰ ثانیه تعیین کرده‌ایم، JMeter پس از ۱۰ ثانیه اول از شروع عملیات تمام کاربرها را راه‌اندازی کرده است. بنابراین در اینجا با ۱۰۰ کاربر و ۱۰ ثانیه Ramp-Up فاصله زمانی بین شروع به کار هر کاربر برابر با ۱۰۰ میلی ثانیه می‌شود. اگر این زمان را با صفر مقدار دهی کنیم JMeter تمام کاربران را در همان لحظه اول راه‌اندازی می‌کند.



شکل ۴-۴ افزودن کاربران به طرح آزمون

Thread Group	
Name:	SwimDB Users
Comments:	
Action to be taken after a Sampler error	
<input type="radio"/> Continue <input checked="" type="radio"/> Start Next Thread Loop <input type="radio"/> Stop Thread <input type="radio"/> Stop Test <input type="radio"/> Stop Test Now	
Thread Properties	
Number of Threads (users):	1000
Ramp-Up Period (in seconds):	10
Loop Count:	<input type="checkbox"/> Forever 10
<input type="checkbox"/> Delay Thread creation until needed	
<input type="checkbox"/> Scheduler	
Scheduler Configuration	
Duration (seconds)	
Startup delay (seconds)	
Start Time	2017/05/11 22:43:55
End Time	2017/05/11 22:43:55

شکل ۴-۵ گزینه‌های Thread Group



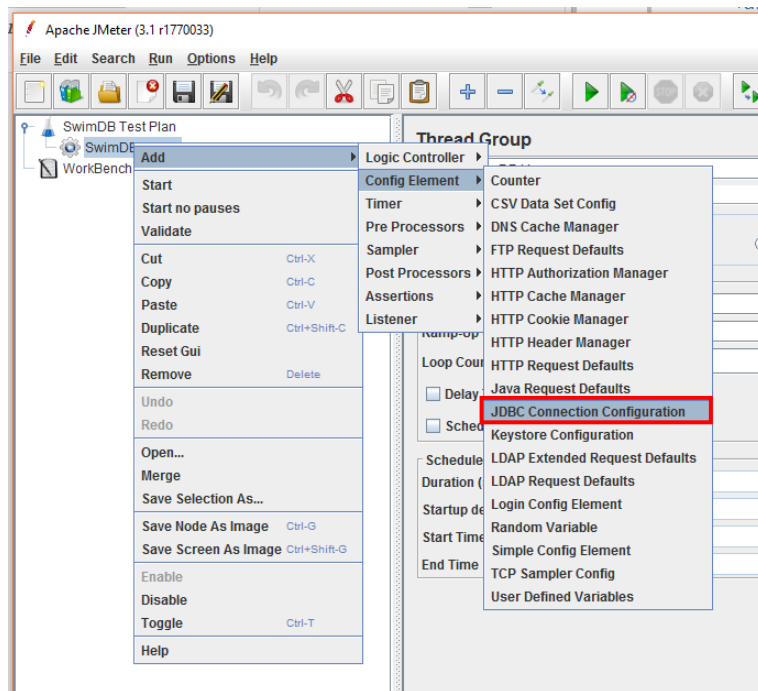
شکل ۴-۶ تنظیمات کنترل پنل کاربران

۳-۲-۴ افزودن درخواست پرس‌وجوی پایگاه داده

اکنون فرمان‌های SQL که قرار است کاربران فرضی ما به سرویس‌دهنده پایگاه داده بفرستند و پاسخ بگیرند را اضافه می‌کنیم. همان طور که در بخش سناریوی آزمون گفتیم دو پرس‌وجوی مهم مد نظر ما هستند. یک پرس‌وجوی مشاهده نتایج انفرادی و یک پرس‌وجوی مشاهده نتایج تیمی برای یک یا مجموعه‌ای از مسابقات.

برای اضافه کردن یک ارتباط به پایگاه داده بر روی عنصر ThreadGroup ایجاد شده در مرحله قبل راست کلیک می‌کنیم و گزینه JDBC Connection Configuration را از مسیر ذیل انتخاب می‌کنیم.

Add → Config Element → JDBC Connection Configuration

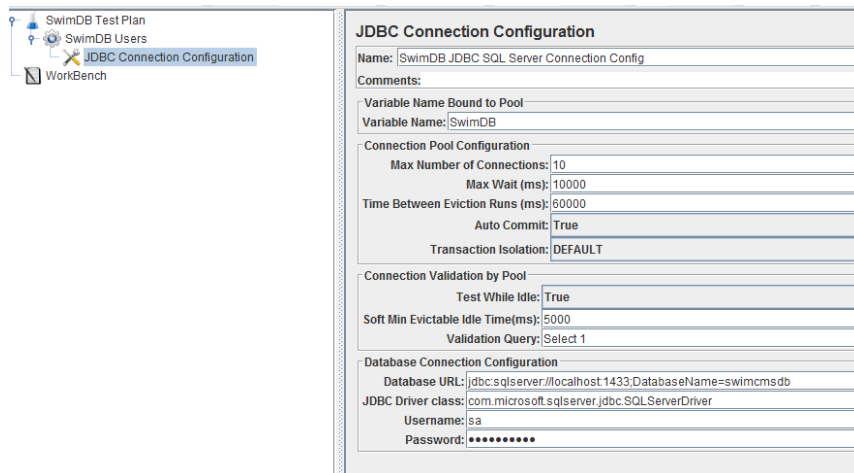


شکل ۴-۷ اضافه کردن یک اتصال JDBC برای ارسال پرس و جوهای پایگاه داده

در پنجره ظاهر شده فیلدهای زیر را تکمیل می‌کنیم.

- متغیر نام: این متغیر یک نام یکتا را مشخص می‌کند که پیکربندی ما مقید می‌شود.
- آدرس پایگاه داده (رشته اتصال): نام سرور پایگاه داده، شماره پورت و نام پایگاه داده‌ای که می‌خواهیم به آن متصل شویم را مشخص می‌کند.
- کلاس درایور JDBC: نام کلاس جاوایی را که برای اتصال به پایگاه داده از آن استفاده می‌شود را مشخص می‌کند.
- نام کاربری: نام کاربری کاربر پایگاه داده را مشخص می‌کند.
- گذرواژه: گذرواژه کاربر را مشخص می‌کند.

بقیه فیلدها را می‌توان با مقادیر پیش فرض رها کرد.



شکل ۴-۸ تنظیمات اتصال شامل رشته اتصال، درایور، شناسه و گذرواژه مدیر پایگاه

رشته اتصال و کلاس درایور برای پایگاه داده‌های مختلف متفاوت است. این مقادیر برای برخی از پایگاه داده‌های شناخته در جدول ۴-۱ نشان داده شده است.

جدول ۴-۱ رشته اتصال و کلاس درایور برای ارتباط JMeter با پایگاه داده از طریق JDBC برای برخی پایگاه داده‌های شناخته شده

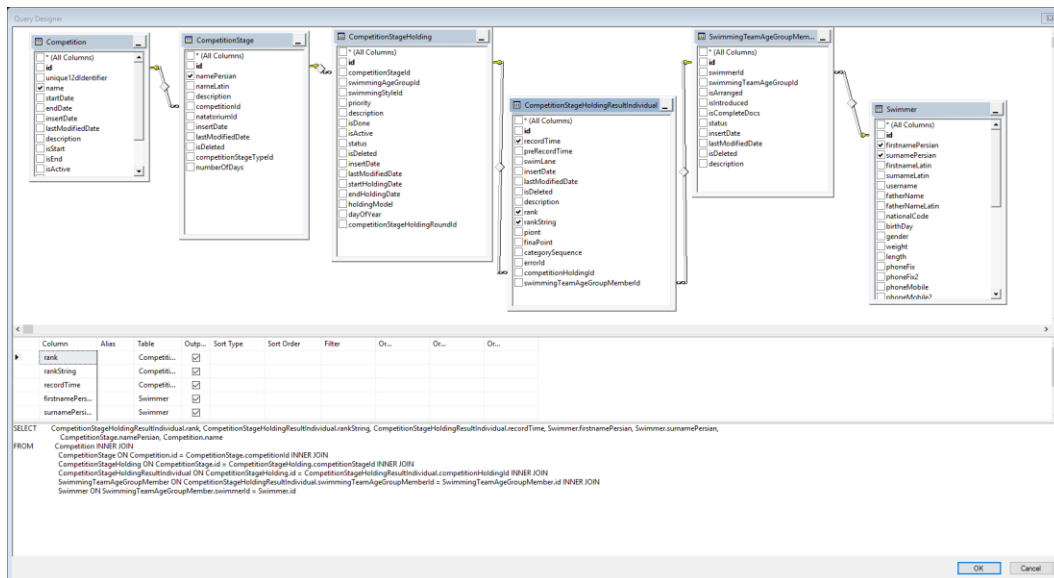
Database	Database URL	JDBC Driver class
MySQL	jdbc:mysql://host[:port]/dbname	com.mysql.jdbc.Driver
PostgreSQL	jdbc:postgresql:{dbname}	org.postgresql.Driver
Oracle	jdbc:oracle:thin:@//host:port/service	oracle.jdbc.OracleDriver
SQL Server (MS JDBC Driver)	jdbc:sqlserver://host:port;DatabaseName=dbname	com.microsoft.sqlserver.jdbc.SQLServerDriver
Apache Derby	jdbc:derby://server[:port]/databaseName[;URLAttributes=value[;...]]	org.apache.derby.jdbc.ClientDriver

اکنون باید پرس‌وجوی خود را وارد کنیم. برای طراحی پرس‌وجوی مد نظر از محیط Microsoft SQL Server Management Studio 2016 استفاده می‌کنیم. نیازی با پاسخ دقیق پرس‌وجو

نداریم و به همین جهت فقط یک پرس‌وجوی کلی را می‌نویسیم. نحوه ساخت پرس‌وجو در شکل ۹-۴ و شکل ۱۱-۴ آمده است.

برای اضافه کردن پرس و جو به JMeter بر روی Thread Group کلیک راست کرده و گزینه JDBC Request را از مسیر زیر انتخاب می‌کنیم.

Add → Sampler → JDBC Request



شکل ۹-۴ نحوه ساخت پرس و جوی مربوط به نشان دادن نتایج انفرادی در محیط SQL Server Management Studio

ما در طرح آزمون خود دو درخواست پرس و جو بر روی پایگاه داده داریم. به روشی که گفته شد این دو درخواست را به طرح آزمون اضافه می‌کنیم. پس از نوشتن پرس‌وجو آن را در قسمت SQL Query وارد می‌کنیم. شکل ۱۰-۴ این عمل را نشان می‌دهد. مراحل طی شده را مجدداً برای اضافه کردن پرس‌وجوی دوم طی می‌کنیم. این مراحل نیز در شکل ۱۱-۴ و ۱۲-۴ نشان داده شده است.

JDBC Request

Name: Individual Result Query
 Comments: Query to show individual result of a holded competition

Variable Name Bound to Pool
 Variable Name: SwimDB

SQL Query

Query Type: Select Statement

```

1 SELECT CompetitionStage.id AS EXPRI,
2 CompetitionStageHoldingResultIndividual.recordTime,
3 CompetitionStageHoldingResultIndividual.swimLane,
4 CompetitionStageHoldingResultIndividual.rank,
5 Swimmer.firstnamePersian,
6 Swimmer.surnamePersian,
7 Competition.name
8 FROM CompetitionStageHoldingResultIndividual
9 INNER JOIN CompetitionStageHolding
10 ON CompetitionStageHoldingResultIndividual.competitionHoldingId = CompetitionStageHolding.id
11 INNER JOIN CompetitionStage
12 ON CompetitionStageHolding.competitionStageId = CompetitionStage.id
13 INNER JOIN Competition
14 ON CompetitionStage.competitionId = Competition.id
15 INNER JOIN Swimmer
16 ON CompetitionStageHoldingResultIndividual.id = Swimmer.id
17 INNER JOIN SwimmingTeam
18 ON CompetitionStageHoldingResultIndividual.id = SwimmingTeam.id
19 ORDER BY recordTime;
    
```

Parameter values:
 Parameter types:
 Variable names:
 Result variable name:
 Query timeout (s):
 Handle ResultSet: Store as String

شکل ۴-۱۰ اضافه کردن یک درخواست پرس و جو به طرح آزمون JMeter

The screenshot shows the Query Designer interface in SQL Server Enterprise Manager. It displays a query graph with the following tables and their relationships:

- Competition**: Contains columns like id, uniqueIdentifier, name, startDate, endDate, description, isActive, etc.
- CompetitionStage**: Contains columns like id, namePersian, nameLatin, description, competitionId, etc.
- CompetitionStageHolding**: Contains columns like id, competitionStageId, swimmingAgeGroupId, swimmingTeamAge, priority, description, etc.
- CompetitionStageHoldingResultTeam**: Contains columns like id, recordTime, preferredTime, swimLane, startDate, endDate, rank, etc.
- SwimmingTeamAgeGroupMember**: Contains columns like id, swimmerId, swimmingTeamAgeGroup, subchanged, etc.
- Swimmer**: Contains columns like id, firstnamePersian, surnamePersian, surnameLatin, etc.

The query graph shows the following relationships:

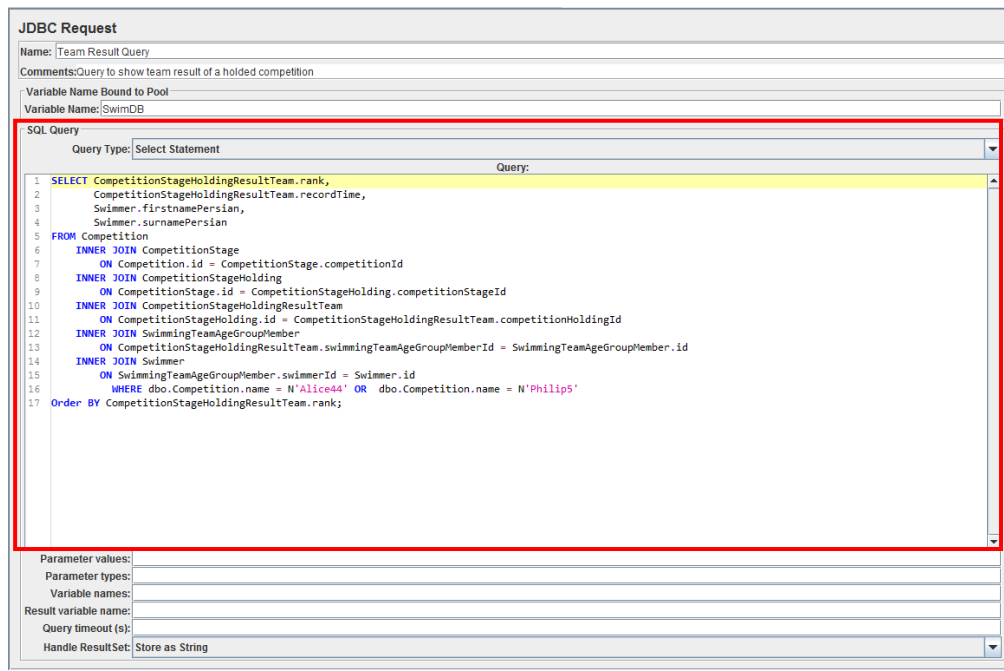
- CompetitionStageHoldingResultTeam is linked to CompetitionStageHolding (via competitionHoldingId).
- CompetitionStageHolding is linked to CompetitionStage (via competitionStageId).
- CompetitionStageHoldingResultTeam is linked to Swimmer (via swimmerId).
- SwimmingTeamAgeGroupMember is linked to CompetitionStageHoldingResultTeam (via id).
- SwimmingTeamAgeGroupMember is linked to Swimmer (via swimmerId).

The query text at the bottom of the window is:

```

SELECT FROM CompetitionStageHoldingResultTeam, CompetitionStageHoldingResultTeam, Swimmer.firstnamePersian, Swimmer.surnamePersian
Competition INNER JOIN
CompetitionStage ON CompetitionId = CompetitionStage.competitionId INNER JOIN
CompetitionStageHolding ON CompetitionStageId = CompetitionStageHolding.competitionStageId INNER JOIN
CompetitionStageHoldingResultTeam ON CompetitionStageHoldingId = CompetitionStageHoldingResultTeam.competitionHoldingId INNER JOIN
SwimmingTeamAgeGroupMember ON CompetitionStageHoldingResultTeam.swimmingTeamAgeGroupMemberId = SwimmingTeamAgeGroupMember.id INNER JOIN
Swimmer ON SwimmingTeamAgeGroupMember.swimmerId = Swimmer.id
GROUP BY CompetitionStageId
    
```

شکل ۴-۱۱ نحوه ساخت پرس و جوی مربوط به نشان دادن نتایج تیمی در محیط SQL Server Management Studio



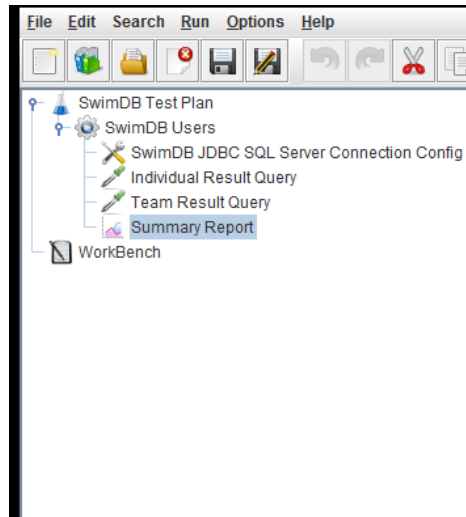
شکل ۴-۱۲ اضافه کردن درخواست پرس و جوی دوم به طرح آزمون JMeter

۴-۲-۴ افزودن عنصر Listener و مشاهده نتایج آزمون

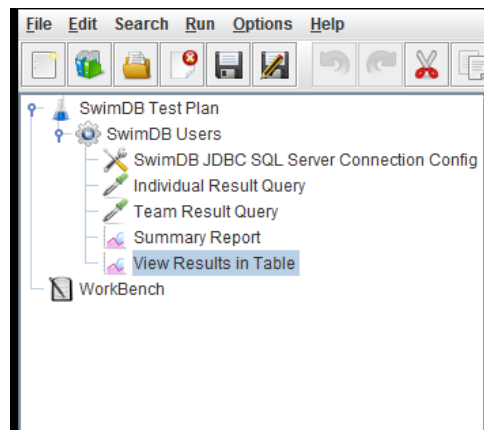
آخرین عنصری که لازم است به طرح آزمون اضافه شود یک عنصر از نوع Listener است. در واقع این عنصر برای مشاهده نتایج آزمون استفاده می‌شود. برای اضافه کردن یک listener بروی Thread Group راست کلیک کرده و از مسیر زیر گزینه Summary Report را انتخاب می‌کنیم.

Add → Listener → Summary Report

Summary Report اطلاعات آزمون انجام شده را به صورت خلاصه نشان می‌دهد. برای مشاهده جزئیات آزمون انجام شده می‌توان از عنصر View Results in Table استفاده کرد. برای اضافه کردن این عنصر نیز به همان روش فوق عمل می‌کنیم. اکنون طرح آزمون ما تکمیل و آماده اجراست.



شکل ۴-۱۳ افزودن یک عنصر Listener برای مشاهده نتایج آزمون



شکل ۴-۱۴ مشاهده نتایج به صورت جدولی

۵ اجرای آزمون و تحلیل نتایج

۱-۵ آزمایش اول

اکنون همه مراحل تهیه طرح آزمون انجام شده است و می‌توانیم آزمون بار را اجرا کنیم. برای این منظور از نوار ابزار بالای برنامه گزینه Start را می‌زنیم. سناریو به همان صورتی که مورد نظر ما بود انجام می‌شود. در آزمایش ما این سناریو به راحتی و مجموع مدت زمان ۹ ثانیه به طور کامل انجام شد.

برای مشاهده خلاصه نتایج آزمایش انجام شده بر روی عنصر Summary Report که در بخش قبل به طرح آزمون خود اضافه کردیم، کلیک می‌نماییم. همان‌طور که در شکل ۵-۱ مشاهده می‌شود، در مجموع ۱۰۰۰ درخواست انجام شده که طبق سناریو ما این مقدار قبلاً محاسبه شده بود. درصد درخواست‌های خراب صفر است که نشان می‌دهد همه درخواست‌ها به درستی انجام شده است. گذردهی کل برابر ۱۰۰,۳ درخواست در ثانیه است که نشان می‌دهد سیستم در حالتی است که بار کاری سنگینی را تحمل نمی‌کند.

Summary Report										
Name: Summary Report										
Comments:										
Write results to file / Read from file										
Filename										Configure
Log/Display Only: <input type="checkbox"/> Errors <input type="checkbox"/> Successes										
Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/sec	Sent KB/sec	Avg. Bytes
Individual Result...	500	12	10	158	7.17	0.00%	50.2/sec	8198.53	0.00	167335.0
Team Result Qu...	500	3	2	10	0.75	0.00%	51.0/sec	8.56	0.00	172.0
TOTAL	1000	8	2	158	6.79	0.00%	100.3/sec	8204.49	0.00	83753.5

شکل ۵-۱ گزارش خلاصه جزئیات آزمون برگزار شده

می‌توان جزئیات دقیق هر درخواست را با کلیک بر روی عنصر View Result in Table مشاهده کرد. این نتایج در شکل ۵-۲ نشان داده شده است.

View Results in Table

Name: /View Results in Table

Comments:

Write results to file / Read from file

Filename: Browse... Log/Display Only: Errors Successes

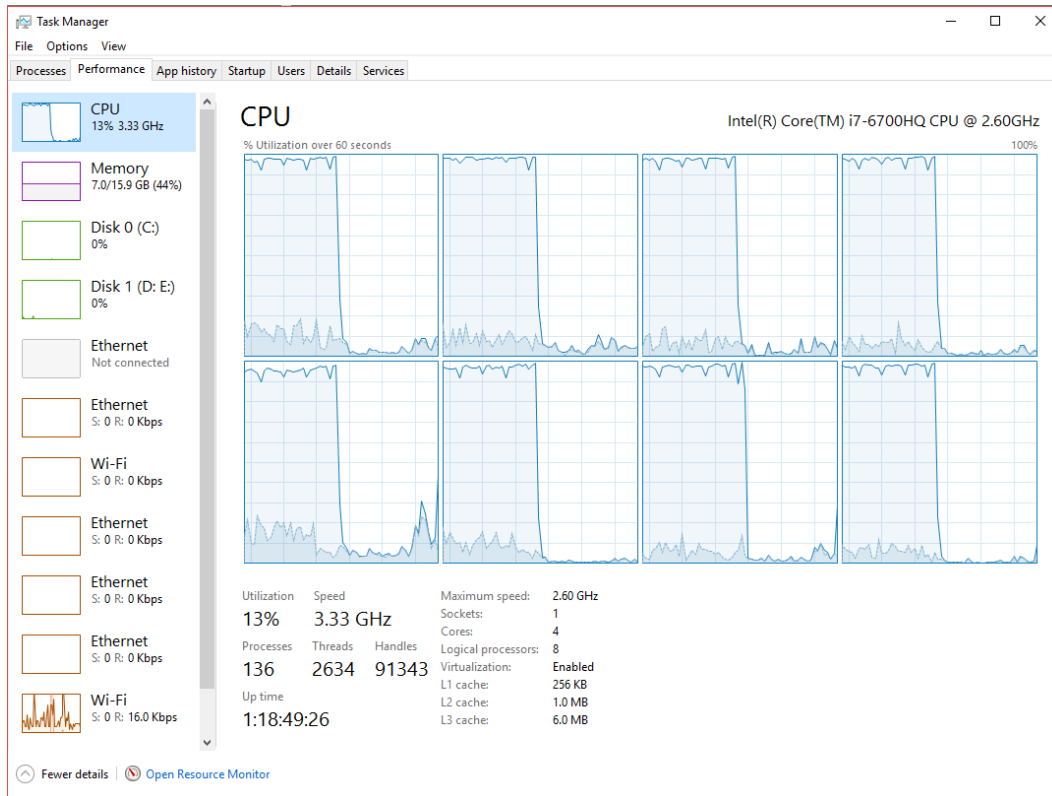
Sample #	Start Time	Thread Name	Label	Sample Time(ms)	Status	Bytes	Sent Bytes	Latency	Connect Time(ms)
1	13:53:20.598	SwimDB Users 1-1	Individual Result ...	158	✓	167335	0	149	115
2	13:53:20.600	SwimDB Users 1-2	Individual Result ...	67	✓	167335	0	56	23
3	13:53:20.666	SwimDB Users 1-1	Team Result Query	5	✓	172	0	5	0
4	13:53:20.667	SwimDB Users 1-2	Team Result Query	7	✓	172	0	7	0
5	13:53:20.671	SwimDB Users 1-1	Individual Result ...	25	✓	167335	0	15	0
6	13:53:20.674	SwimDB Users 1-2	Individual Result ...	24	✓	167335	0	16	0
7	13:53:20.696	SwimDB Users 1-1	Team Result Query	5	✓	172	0	5	0
8	13:53:20.698	SwimDB Users 1-2	Team Result Query	6	✓	172	0	6	0
9	13:53:20.704	SwimDB Users 1-2	Individual Result ...	13	✓	167335	0	9	0
10	13:53:20.700	SwimDB Users 1-3	Individual Result ...	19	✓	167335	0	14	0
11	13:53:20.717	SwimDB Users 1-2	Team Result Query	3	✓	172	0	3	0
12	13:53:20.701	SwimDB Users 1-1	Individual Result ...	22	✓	167335	0	19	0
13	13:53:20.719	SwimDB Users 1-3	Team Result Query	5	✓	172	0	5	0
14	13:53:20.724	SwimDB Users 1-1	Team Result Query	4	✓	172	0	4	0
15	13:53:20.720	SwimDB Users 1-2	Individual Result ...	13	✓	167335	0	9	0
16	13:53:20.733	SwimDB Users 1-2	Team Result Query	3	✓	172	0	3	0
17	13:53:20.724	SwimDB Users 1-3	Individual Result ...	20	✓	167335	0	16	0
18	13:53:20.744	SwimDB Users 1-3	Team Result Query	8	✓	172	0	8	0
19	13:53:20.736	SwimDB Users 1-2	Individual Result ...	17	✓	167335	0	8	0
20	13:53:20.728	SwimDB Users 1-1	Individual Result ...	26	✓	167335	0	17	0
21	13:53:20.753	SwimDB Users 1-2	Team Result Query	4	✓	172	0	4	0
22	13:53:20.754	SwimDB Users 1-1	Team Result Query	10	✓	172	0	10	0
23	13:53:20.753	SwimDB Users 1-3	Individual Result ...	17	✓	167335	0	13	0
24	13:53:20.770	SwimDB Users 1-3	Team Result Query	8	✓	172	0	8	0
25	13:53:20.794	SwimDB Users 1-1	Individual Result ...	17	✓	167335	0	11	0
26	13:53:20.781	SwimDB Users 1-1	Team Result Query	9	✓	172	0	9	0
27	13:53:20.778	SwimDB Users 1-3	Individual Result ...	16	✓	167335	0	12	0
28	13:53:20.794	SwimDB Users 1-3	Team Result Query	4	✓	172	0	4	0
29	13:53:20.798	SwimDB Users 1-3	Individual Result ...	16	✓	167335	0	13	0
30	13:53:20.801	SwimDB Users 1-4	Individual Result ...	18	✓	167335	0	14	0
31	13:53:20.814	SwimDB Users 1-3	Team Result Query	6	✓	172	0	6	0
32	13:53:20.819	SwimDB Users 1-4	Team Result Query	4	✓	172	0	4	0
33	13:53:20.823	SwimDB Users 1-4	Individual Result ...	12	✓	167335	0	8	0
34	13:53:20.835	SwimDB Users 1-4	Team Result Query	3	✓	172	0	3	0

Scroll automatically? Child samples? No of Samples 1000 Latest Sample 3 Average 8 Deviation 6

شکل ۲-۵ جزئیات کامل درخواست‌های فرستاده شده و پاسخ‌های دریافتی در طی آزمون

۲-۵ آزمایش دوم

در آزمایش اول تعداد ۱۰۰۰ درخواست را به سمت سرویس دهنده پایگاه داده فرستادیم و سیستم بار کاری سنگینی را متحول نشد. در این آزمایش تعداد کاربران را به ۱۰۰۰ و تعداد درخواست ها را به ۱۰ افزایش می‌دهیم و مجدداً طرح آزمون را اجرا می‌کنیم. نتایج آزمایش دوم در شکل ۳-۵ و شکل ۴-۵ نشان داده شده است.



شکل ۳-۵ وضعیت پردازنده مرکزی در حین انجام آزمایش دوم

Summary Report

Name: Summary Report

Comments:

Write results to file / Read from file

Filename: Log/Display Only: Errors Successes

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/sec	Sent KB/sec	Avg. Bytes
Individual Result...	10500	1907	10	10021	1225.49	0.02%	33.6/sec	5487.72	0.00	167303.1
Team Result Qu...	10498	1904	2	10014	1231.43	0.05%	33.6/sec	5.64	0.00	171.9
TOTAL	20998	1906	2	10021	1228.46	0.03%	67.2/sec	5493.29	0.00	83745.5

شکل ۴-۵ خلاصه نتایج آزمایش دوم

۶ مراجع

- [1] “Building a Database Test Plan,” *Apache*, 2017. [Online]. Available: <http://jmeter.apache.org/usermanual/build-db-test-plan.html>.
- [2] “Database Testing,” 2017. [Online]. Available: https://www.tutorialspoint.com/database_testing/database_testing_performance.htm.
