Oracle 23c Free with AI Vector Search
Limited Availability Program

Installation and Configuration

Version 1.3

Doug Hood, 2023

This document covers the following:

- How to create VMs for Oracle 23c Free
- How to install the vector enabled Oracle 23c Free on Oracle Cloud & AWS
    - Installing the prerequisite RPM
    - Installing the Oracle 23c Free RPM
    - Configuring Oracle 23c Free
- How to create a PDB user
    - Creating a PDB tablespace
    - Creating a vector schema user
    - Optionally creating a PDB service
    - Create tables with vectors
- How to install and configure python-oracledb on Oracle Linux 8.7 and 8.8
- How to install and configure python-oracledb on Oracle Linux 8.6
- How to install and configure node-oracledb on Oracle Linux 8.6, 8.7 and 8.8
- How to install and configure JDK11, 21 and OpenJDK 17 on Oracle Linux 8.6, 8.7 and 8.8
- How to install and configure JDK8 on Oracle Linux 8.8
- How to use OpenAI vector embeddings with python-oracledb
- How to use Cohere vector embeddings with python-oracledb
- How to use Sentence Transformers vector embeddings with python-oracledb
- How to use OpenAI vector embeddings with node-oracledb
- How to use Cohere vector embeddings with node-oracledb

The vector enabled version of Oracle 23c Free supports Oracle Linux 8.6, 8.7, 8.8 and 8.9 for x8664. Oracle 23c Free will not work with Oracle Linux 7.9, 8.5 or 9.x.
You need at least:

- **30 GB of disk space** [due to the dependencies for the sample code]
- **2 GB RAM. [8+ GB RAM for the Linux host is recommended]**

On AWS, the following VMs have been tested with vectors on Oracle 23c Free:

- AlmaLinux **8.9** 20231123 with t2.micro using **1 GB RAM** and 30 GB storage
  - There is insufficient RAM, so the install will fail
- AlmaLinux **8.9** 20231123 with t2.small using **2 GB RAM** and 30 GB storage
  - The install succeeds, but the system is very slow
- AlmaLinux **8.9** 20231123 with t2.xlarge using **16 GB RAM** and 30 GB storage

On Oracle Cloud, the following VMs have been tested with vectors on Oracle 23c Free:

- Oracle Linux 8.8, image build: 2023.10.24-0
- AlmaLinux OS 8 (x86_64), **8.6**.20230525
- AlmaLinux OS 8 (x86_64), **8.7**.20230525
- AlmaLinux OS 8 (x86_64), **8.8**.20230525
- The above VMs worked with 1 OCPU, 12 GB RAM and the default [30 GB] block storage
- Oracle Linux 8.8 with:
  - 12, 16, 256, 512 and 768 GB RAM are OK
  - 1024 and 2048 GB RAM **fails during the configuration step. This is a bug**
    - Do not install on a machine with more than 768 GB RAM

How to create an OCI or AWS VMs for Oracle 23c Free

- There is nothing special about creating the VMs
- You just need sufficient RAM and disk storage

The instructions to install/configure are the same for Oracle Cloud and AWS.

Install Oracle 23c Free on an Oracle Linux 8.6+ OCI VM

If needed, refer to the Oracle 23c Free Install Guide.

Which Linux OS user to use when

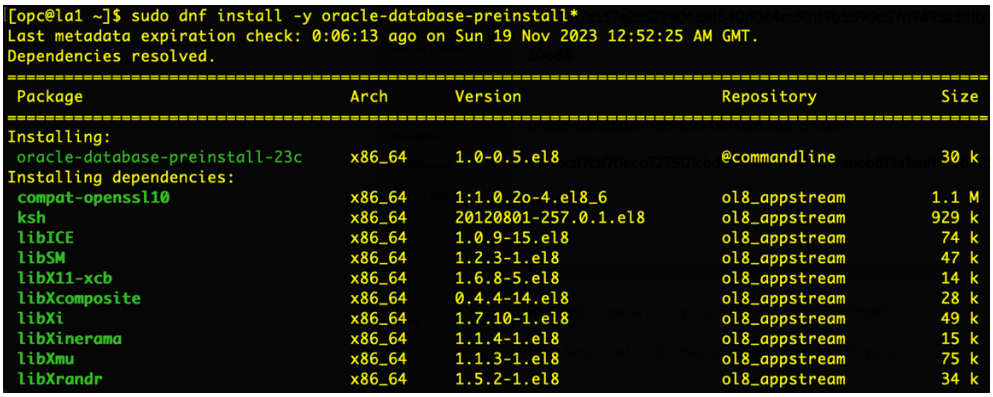
- Some of the operations to install the software require a non-oracle user which has the **sudo** privilege
- All of the Oracle database operations [eg sqlplus, running programs with python-oracledb, node-oracledb and JDBC] should be done with the Linux **oracle** user
- The instructions tell you when to switch between the Linux OS users as needed.

Unzip the software

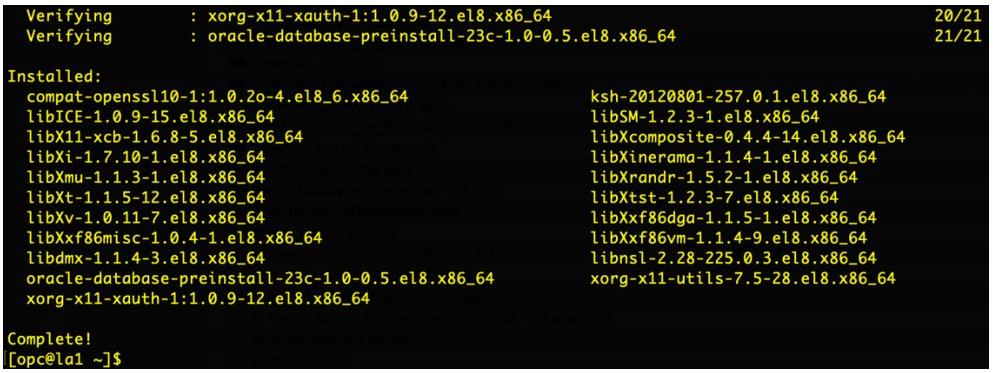
```
unzip p36050269_23.0.0.0.0_Linux-x86-64.zip
```

Pre-install the required dependencies as the non-oracle **user**

```
sudo dnf install -y oracle-database-preinstall*
```



...



Install the database

```
sudo dnf install -y oracle-database-free*
```

```
[opc@la1 ~]$
[opc@la1 ~]$ sudo dnf install -y oracle-database-free*
Last metadata expiration check: 0:08:13 ago on Sun 19 Nov 2023 12:52:25 AM GMT.
Dependencies resolved.
================================================================================
 Package                    Architecture    Version          Repository      Size
================================================================================
Installing:
 oracle-database-free-23c   x86_64          1.0-1            @commandline    1.6 G

Transaction Summary
================================================================================
Install  1 Package

Total size: 1.6 G
Installed size: 4.1 G
Downloading Packages:
Running transaction check
Transaction check succeeded.
```

…

```
Transaction test succeeded.
Running transaction
  Preparing        :                                                        1/1
  Running scriptlet: oracle-database-free-23c-1.0-1.x86_64                  1/1
  Installing       : oracle-database-free-23c-1.0-1.x86_64                  1/1
  Running scriptlet: oracle-database-free-23c-1.0-1.x86_64                  1/1
[INFO] Executing post installation scripts...
[INFO] Oracle home installed successfully and ready to be configured.
To configure Oracle Database Free, optionally modify the parameters in '/etc/sysconfig/oracle-free-23c.
conf' and then run '/etc/init.d/oracle-free-23c configure' as root.

  Verifying        : oracle-database-free-23c-1.0-1.x86_64                  1/1

Installed:
  oracle-database-free-23c-1.0-1.x86_64

Complete!
[opc@la1 ~]$
```

Configure the database

```
sudo /etc/init.d/oracle-free-23c configure
```

```
[opc@la1 ~]$ sudo /etc/init.d/oracle-free-23c configure
Specify a password to be used for database accounts. Oracle recommends that the password entered should
 be at least 8 characters in length, contain at least 1 uppercase character, 1 lower case character and
 1 digit [0-9]. Note that the same password will be used for SYS, SYSTEM and PDBADMIN accounts:
Confirm the password:
Configuring Oracle Listener.
Listener configuration succeeded.
Configuring Oracle Database FREE.
Enter SYS user password:
**********
Enter SYSTEM user password:
*********
Enter PDBADMIN User Password:
*********
Prepare for db operation
7% complete
Copying database files
29% complete
Creating and starting Oracle instance
30% complete
33% complete
```

…

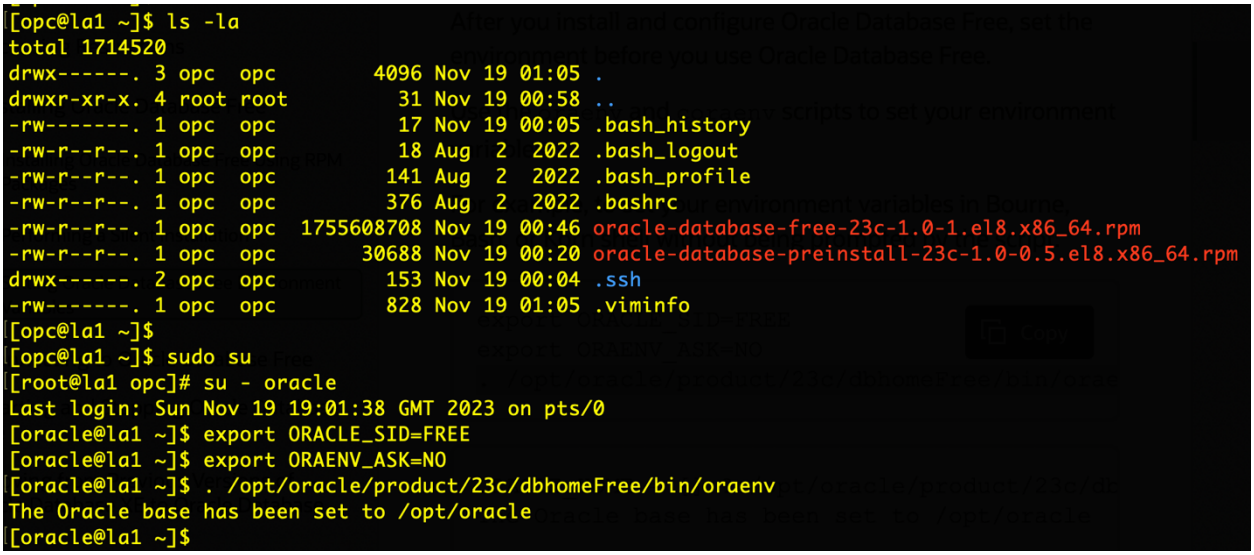
```
49% complete
50% complete
Creating Pluggable Databases
54% complete
71% complete
Executing Post Configuration Actions
93% complete
Running Custom Scripts
100% complete
Database creation complete. For details check the logfiles at:
 /opt/oracle/cfgtoollogs/dbca/FREE.
Database Information:
Global Database Name:FREE
System Identifier(SID):FREE
Look at the log file "/opt/oracle/cfgtoollogs/dbca/FREE/FREE.log" for further details.

Connect to Oracle Database using one of the connect strings:
     Pluggable database: la1/FREEPDB1
     Multitenant container database: la1
[opc@la1 ~]$
```

Change to the Linux oracle user & set the bash environment for the oracle Linux user

sudo su

su - oracle

export ORACLE_SID=FREE
export ORAENV_ASK=NO
. /opt/oracle/product/23c/dbhomeFree/bin/oraenv



Connect as the default Oracle 23c Free PDB service & create a tablespace and user

sqlplus sys@localhost:1521/freepdb1 as sysdba

#
Make sure to use **single quotes** around the **tbs1.dbf** datafile name.
#
create tablespace tbs1
datafile '**tbs1.dbf**' size 1G
extent management local
segment space management auto;

create user vector identified by vector
default tablespace tbs1
quota unlimited on tbs1;

grant DB_DEVELOPER_ROLE to vector;

exit

```
[oracle@la1 ~]$ sqlplus sys@localhost:1521/FREEPDB1 as sysdba

SQL*Plus: Release 23.0.0.0.0 - Development on Sun Nov 19 21:51:15 2023
Version 23.4.0.23.11

Copyright (c) 1982, 2023, Oracle.  All rights reserved.

Enter password:

Connected to:
Oracle Database 23c Free Release 23.0.0.0.0 - Develop, Learn, Run for Free (LA)
Version 23.4.0.23.11

SQL> CREATE TABLESPACE tbs1
   DATAFILE 'tbs.dbf' SIZE 1G
   EXTENT MANAGEMENT LOCAL
   SEGMENT SPACE MANAGEMENT AUTO;  2    3    4

Tablespace created.

SQL> create user vector identified by vector
DEFAULT TABLESPACE tbs1
quota unlimited on tbs1;
  2    3
User created.

SQL> grant DB_DEVELOPER_ROLE to vector;

Grant succeeded.

SQL>
SQL> exit
Disconnected from Oracle Database 23c Free Release 23.0.0.0.0 - Develop, Learn, Run for Free (LA)
Version 23.4.0.23.11
```

Create a PDB service [freepdb1] in tnsnames using your favorite editor, eg

```
vi $ORACLE_HOME/network/admin/tnsnames.ora
```

You can copy the FREE entry, call it **freepdb1** and change the SERVICE_NAME to **freepdb1**.

```
FREE =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = la1.sub04012011450.vnc3.oraclevcn.com)(PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = FREE)
    )
  )

freepdb1 =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = la1.sub04012011450.vnc3.oraclevcn.com)(PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = FREEPDB1)
    )
  )
```

Now you can choose to connect as either

```
sqlplus vector/vector@localhost:1521/freepdb1
```

or

```
sqlplus vector/vector@freepdb1
```

```
[oracle@la1 admin]$ sqlplus vector/vector@freepdb1

SQL*Plus: Release 23.0.0.0.0 - Development on Sun Nov 19 23:11:19 2023
Version 23.4.0.23.11

Copyright (c) 1982, 2023, Oracle.  All rights reserved.

Last Successful login time: Sun Nov 19 2023 23:09:48 +00:00

Connected to:
Oracle Database 23c Free Release 23.0.0.0.0 - Develop, Learn, Run for Free (LA)
Version 23.4.0.23.11

SQL> create table v2 (v vector(768, float32));

Table created.

SQL> desc v2;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 V                                                  VECTOR(768, FLOAT32)

SQL> exit
Disconnected from Oracle Database 23c Free Release 23.0.0.0.0 - Develop, Learn, Run for Free (LA)
Version 23.4.0.23.11
[oracle@la1 admin]$
```

Move the directories to the oracle user's home directory

You did the Oracle 23c Free install via sudo [effectively root] from the OCI **opc** user [or equivalent AWS **ec2-user**].
Now you want to run your python-oracledb, node-oracledb and JDBC programs as user **oracle**.
Exit from the current **oracle** shell.

```
exit
```

Exit from the current root shell.

```
exit
```

Zip the docs, drivers and sample code directories as a file called la1.zip.

```
zip -r la1.zip docs drivers sample_code
```

Copy the la1.zip to the /tmp directory

```
cp la1.zip /tmp
```

Become the **oracle** user

```
sudo su
su - oracle
```

Copy the la1.zip file from /tmp to the oracle user's home directory.

```
cp /tmp/la1.zip .
```

Unzip the la1.zip file to the oracle home directory.

```
unzip la1.zip
```

The following instructions are best practice for local installs for the vector enabled python-oracledb driver on Python 3.11 on Oracle Linux 8.7 or 8.8

Install Python 3.11 on the Oracle Linux 8.7 or 8.8 VM

`exit` ← Exit from the current **oracle** shell to use the existing root shell

```
sudo dnf install python3.11
```

As user oracle, make sure that pip is installed

```
su - oracle
python3.11 -m ensurepip
```

As user oracle, upgrade pip to enable the local python-oracledb install to work

```
python3.11 -m pip install --upgrade pip
```

As user Oracle, install the vector enabled python-oracledb driver

```
cd ~/drivers
pip3 install oracledb-2.0.0.dev20231121-cp311-cp311-
manylinux_2_17_x86_64.manylinux2014_x86_64.whl
```

Configure the Oracle environment

```
export ORACLE_SID=FREE
export ORAENV_ASK=NO
. /opt/oracle/product/23c/dbhomeFree/bin/oraenv
export TNS_ADMIN=/opt/oracle/product/23c/dbhomeFree/network/admin
```

Run some python scripts using AI Vector Search

```
cd ~/sample_code/python
python3.11 vec1.py
python3.11 vec2.py
python3.11 vec3.py
```

The following instructions are best practice for local install for the vector enabled python-oracledb driver on Python 3.9 on Oracle Linux 8.6

Install Python 3.9 on the Oracle Linux 8.6 VM

`exit` ← Exit from the current **oracle** shell to use the existing root shell

```
sudo dnf install python3.9
```

As user oracle**, make sure that pip is installed**

```
python3.9 -m ensurepip
```

As user oracle**, upgrade pip to enable the local python-oracledb install to work**

```
python3.9 -m pip install --upgrade pip
```

Install the vector enabled python-oracledb driver

```
pip3 install oracledb-2.0.0.dev20231121-cp39-cp39-
manylinux_2_17_x86_64.manylinux2014_x86_64.whl
```

Configure the Oracle environment

```
export ORACLE_SID=FREE
export ORAENV_ASK=NO
. /opt/oracle/product/23c/dbhomeFree/bin/oraenv
export TNS_ADMIN=/opt/oracle/product/23c/dbhomeFree/network/admin
```

Run a python script using AI Vector Search

```
python3.9 vec1.py
python3.9 vec2.py
python3.9 vec3.py
```

How to use Sentence Transformers embedding models with python-oracledb

As user oracle**, install the Sentence Transformers library**

```
pip install -U sentence-transformers
```

It can take a few minutes to do the initial installs of the various sentence-tranformers libraries used by this sample code.

*You do NOT need an API Key to use the **Sentence Transformers** from Hugging Face.*
The sentence transformers used in these examples have an Apache 2.0 license.

Configure the Oracle environment

```
export ORACLE_SID=FREE
export ORAENV_ASK=NO
. /opt/oracle/product/23c/dbhomeFree/bin/oraenv
export TNS_ADMIN=/opt/oracle/product/23c/dbhomeFree/network/admin
```

Run a python script to create vectors via the Sentence Transformers embedding model

```
python3.11 hf_st_embed1.py
python3.11 hf_st_embed2.py
python3.11 hf_st_embed3.py
```

How to use OpenAI embedding models with python-oracledb

As user oracle**, install the OpenAI library**

```
pip install openai==0.28.0
pip install numpy
```

You need to register with openai.com to get an API Key.
You can get free trial API keys, or paid API keys.
The free API keys limit the number and rate of vector creation.
Both free and paid API keys work the same for this example.
The API key is needed at runtime to authenticate and authorize your use of the embedding API.

Configure the OpenAI environment

```
export OPENAI_API_KEY=sk-s************************FVX
```

Configure the Oracle environment

```
export ORACLE_SID=FREE
export ORAENV_ASK=NO
. /opt/oracle/product/23c/dbhomeFree/bin/oraenv
export TNS_ADMIN=/opt/oracle/product/23c/dbhomeFree/network/admin
```

Run a python script to create vectors via the OpenAI embedding model

```
python3.11 openai_embed1.py
```

How to use Cohere embedding models with python-oracledb

As user oracle**, install the Cohere library**

```
pip install cohere
```

You need to register with cohere.com to get an API Key.
You can get free trial API keys, or paid API keys.
The free API keys limit the number and rate of vector creation.
Both free and paid API keys work the same for this example.
The API key is needed at runtime to authenticate and authorize your use of the embedding API.
The same API key can be used for python-oracledb and node-oracledb.

Configure the Cohere environment

```
export CO_API_KEY=R6*****************************Ku
```

Configure the Oracle environment

```
export ORACLE_SID=FREE
export ORAENV_ASK=NO
. /opt/oracle/product/23c/dbhomeFree/bin/oraenv
export TNS_ADMIN=/opt/oracle/product/23c/dbhomeFree/network/admin
```

Run a python script using to create vectors via the Cohere embedding model

```
python3.11 cohere_embed1.py
python3.11 cohere_embed2.py
python3.11 cohere_embed3.py
python3.11 cohere_embed4.py
```

The following instructions are best practice for a local install for the vector enabled node-oracledb driver on Node 18 on Oracle Linux 8.6, 8.7 and 8.8

Check the supported version of Node.js on your machine

`exit` ← Exit from the current OCI oracle shell to use the existing root shell

```
sudo dnf module list --all nodejs
```

Choose Node 18

```
sudo dnf module enable nodejs:18
```

Install Node 18

```
sudo dnf module install nodejs
```

As user oracle, install vector enabled node-oracledb SQL driver

```
su - oracle
cd ~/sample_code/node
npm install oracledb-6.3.0-vec1.tgz
```

Configure the Oracle environment

```
export ORACLE_SID=FREE
export ORAENV_ASK=NO
. /opt/oracle/product/23c/dbhomeFree/bin/oraenv
```

Run some vector programs

```
node vec1.js
node vec2.js
node vec3.js
node vec4.js
```

How to use Cohere embedding models with node-oracledb

As user oracle, install the Cohere library

```
npm install cohere-js
npm install axios
```

You need to register with cohere.com to get an API Key.
You can get free trial API keys, or paid API keys.
The free API keys limit the number and rate of vector creation.
Both free and paid API keys work the same for this example.
The API key is needed at runtime to authenticate and authorize your use of the embedding API.
The same API key can be used for python-oracledb and node-oracledb.

Configure the Cohere environment

```
export CO_API_KEY=R6********************************vwKu
```

Configure the Oracle environment

```
export ORACLE_SID=FREE
export ORAENV_ASK=NO
. /opt/oracle/product/23c/dbhomeFree/bin/oraenv
export TNS_ADMIN=/opt/oracle/product/23c/dbhomeFree/network/admin
```

Create vectors using the Cohere embedding model

```
node cohere_embed1.js
node cohere_embed2.js
node cohere_embed3.js
node cohere_embed4.js
```

How to use OpenAI embedding models with node-oracledb

As user oracle, install the OpenAI library

```
npm install --save openai
```

You need to register with openai.com to get an API Key.
You can get free trial API keys, or paid API keys.
The free API keys limit the number and rate of vector creation.
Both free and paid API keys work the same for this example.
The API key is needed at runtime to authenticate and authorize your use of the embedding API.
The same API key can be used for python-oracledb and node-oracledb.

Configure the OpenAI environment

```
export OPENAI_API_KEY=sk-****************************FVX
```

Configure the Oracle environment

```
export ORACLE_SID=FREE
export ORAENV_ASK=NO
. /opt/oracle/product/23c/dbhomeFree/bin/oraenv
export TNS_ADMIN=/opt/oracle/product/23c/dbhomeFree/network/admin
```

Create vectors using the OpenAI embedding model

```
node openai_embed.js
```

The following instructions are best practice using vectors with JDBC 4.3 on Oracle Linux 8.8 using JDK11 or JDK21 & OpenJDK 17.09

As user opc or root, check the supported version of Java on your machine

`exit` ← Exit from the current oracle shell to use the existing root shell

```
sudo dnf list jdk*
```

Install the relevant JDK

```
sudo dnf install -y jdk-11.x86_64
or
sudo dnf install -y jdk-21-headful.x86_64

or
yum install java-17-openjdk-devel
```

Become the oracle **user**

```
su - oracle
```

Configure the Oracle and JDK environments

```
export ORACLE_SID=FREE
export ORAENV_ASK=NO
. /opt/oracle/product/23c/dbhomeFree/bin/oraenv
export CLASSPATH=$ORACLE_HOME/jdbc/lib/ojdbc11.jar:.
export TNS_ADMIN=$ORACLE_HOME/network/admin
```

Run some vector programs

```
cd ~/sample_code/jdbc
javac vec1.java
java vec1

javac vec2.java
java vec2

javac vec3.java
java vec3

javac vec4.java
java vec4
```

The following instructions are best practice using vectors with JDBC 4.2 on Oracle Linux 8.8 using JDK8

Check the supported version of Java on your machine

```
sudo dnf list jdk*
```

Install the relevant JDK

```
sudo dnf install -y jdk-1.8.x86_64
```

Configure the Oracle and JDK environments

```
export ORACLE_SID=FREE
export ORAENV_ASK=NO
. /opt/oracle/product/23c/dbhomeFree/bin/oraenv
export CLASSPATH=$ORACLE_HOME/jdbc/lib/ojdbc8.jar:.
export TNS_ADMIN=$ORACLE_HOME/network/admin
```

Run some vector programs

```
javac vec1.java
java vec1

javac vec2.java
java vec2

javac vec3.java
java vec3

javac vec4.java
java vec4
```