PERSONAL
DEVELOPMENT
PROCESS
stay weird, stay different

Java dasturlash tili.
JVM, JRE va JDK.
Javada birinchi dastur.
Intellij IDEA ilovasi.

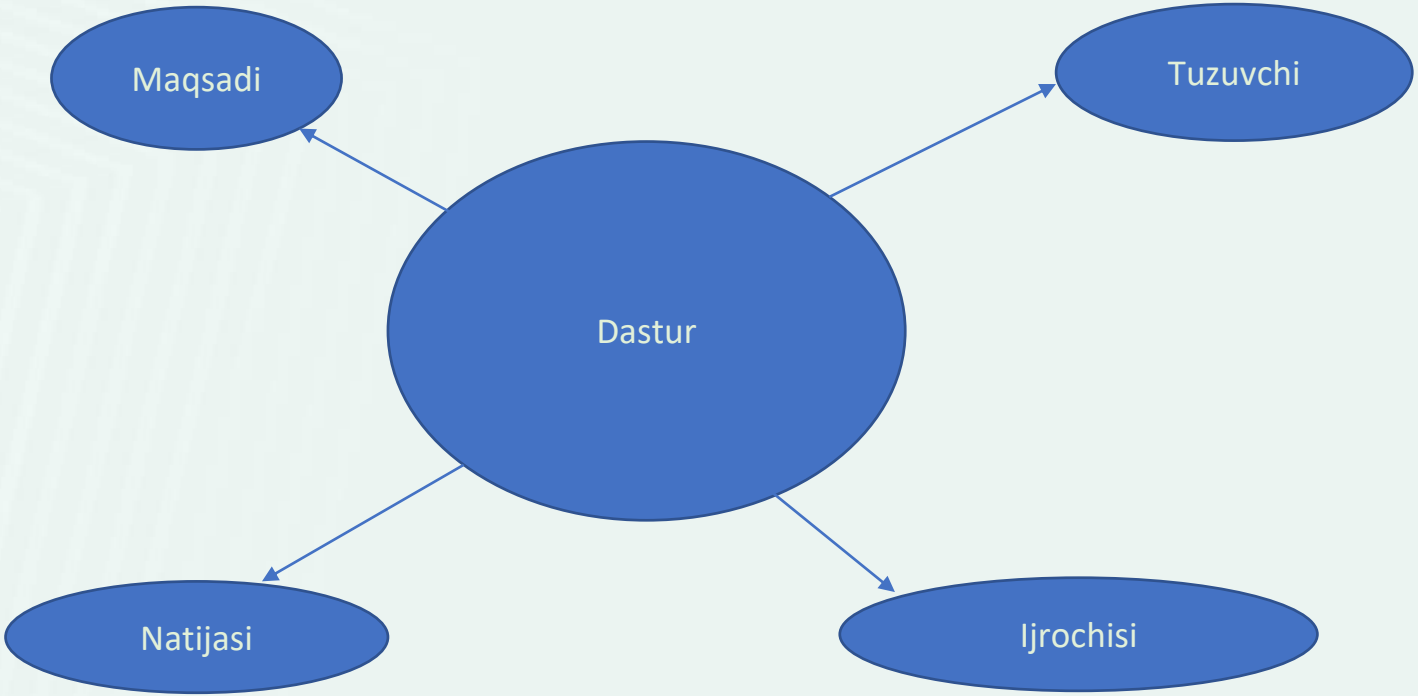
PERSONAL
DEVELOPMENT PROCESS
stay weird, stay different

DASTUR BU NIMA-?

PERSONAL
DEVELOPMENT PROCESS
stay weird, stay different

Dastur deb - ma’lum bir shartlar, o’rnatilgan aniq qoidalar yoki qat’iy belgilangan me’yorlar asosida bajariladigan amallar ketma-ketligiga aytiladi.

Masalan: Davlat dasturi, tv ko’rsatuvlar dasturi, konsert dasturi, o’quv dasturi, komputer dasturi va h.k. .


Maqsadi
Tuzuvchi
Dastur
Natijasi
Ijrochisi

KOMPYUTER DASTURI

PERSONAL
DEVELOPMENT PROCESS
stay weird, stay different

Komputerda bajariladigan amallar ketma-ketligiga komputer dasturi deb aytiladi.

Komputer dasturlari 3 turga bo’linadi:

- 1. **Amaliy dasturlar** - foydalanuvchi bevosita ishlashi uchun mo’ljallangan dasturlar, masalan matn va rasm muharrirlari va hokazo.
- 2. **Tizimli dasturlar** - komp’yuter qurilmalarining ishchi holatini nazorat qiluvchi va boshqaruvchi dasturlar (Operatsion sistemalar, drayverlar, …)
- 3. **Uskunaviy tizimlar** — komp’yuter uchun yangi dasturlar tuzishni ta’minlash tizimlari (dasturlash tillari, IDEA…)

DASTURLASH TILLARI

- Dasturlash tillari 2 ta katta guruhlarga bo'linadi, Quyi va Yuqori darajali dasturlash tili. Quyi darajali dasturlash tili ancha murakkab bo'lib ular juda maxsus sohalarda ishlatiladi va ularning mutaxassislari ham juda kam. Chunki quyi dasturlash tillari (masalan: assembler) ko'pincha mikroprotsessorlar bilan ishlashda kerak bo'lishi mumkin. Asosan turli dasturlash ishlari uchun yuqori darajali dasturlash tilidan keng foydalaniladi.

- Odatda dasturlash yuqori darajali dasturlash tillari (Java, C++, C#, Python, Clojure,…) vositasida amalga oshiriladi. Bu dasturlash tillarining semantikasi odam tiliga yaqinligi tufayli dastur tuzish jarayoni ancha oson kechadi. Ular yordamida kod insonga "tushunarli" tilda yoziladi. Ingliz tilini yaxshi biluvchilar programma kodini qiynalmasdan tushunishlari mumkin.

JAVA DASTURLASH TILI

Java dasturlash tili - eng ommaviy dasturlash tillaridan biri bo'lib, Oak dasturlash tili asosida paydo bo'lgan. Oak dasturlash tili 90-yillarning boshida Sun Microsystems tomonidan platformaga(Operatsion tizimga) bog'liq bo'lmagan holda ishlovchi yangi avlod aqlli qurilmalarini yaratishni maqsad qilib harakat boshlagan edi. Bunga erishish uchun Sun Microsystems hodimlari C++ ni ishlatishni rejalashtirdilar, lekin ba'zi sabablarga ko'ra bu fikridan voz kechishdi. Oak muvofaqiyatsiz chiqdi va 1995-yilda Sun uning nomini Java ga almashtirdi, va uni WWW rivojlanishiga hizmat qilishi uchun ma'lum o'zgarishlar qilishdi. Java Obyektga Yo'naltirilgan Dasturlash(OOP-object oriented programming) tili va u C++ ga ancha o'xshash. Eng ko'p yo'l qo'yildigan xatolarga sabab bo'luvchi qismalari olib tashlanib, Java dasturlash tili ancha soddalashtirildi. Javada kod yozilgan fayllar(*.java bilan nihoyalanuvchi) kompilatsiyadan keyin bayt kod(*.class bilan nomlanuvchi bytecode) ga o'tadi va bu bayt kod interpretator tomonidan o'qib yurgizdiriladi.

- **Object Oriented** – Javada hamma narsa ob’ekt shaklidadir. Javani ob'ekt modeliga asoslanganligi tufayli osongina kengaytirish mumkin.
- **Platform Independent** – Javada kompilyatsiya boshqa dasturlash tillaridagi kabi (C, C++ ..) aynan biror-bir platforma uchun emas balki platformalarga bog’liq bo’lmagan bayt-kodga kompilyatsiya bo’ladi. Bayt kod esa JVM o’rnatilgan ixtiyoriy qurilmada ishga tushishi mumkin.
- **Simple** – Java o’rganishga oson qilib yaratilgan. Agarda siz OOP ning asosiy konsepsiyalarni bilsangiz javani o’zlashtirish siz uchun qiyin bo’lmaydi.
- **Secure** – himoyalanganlik funksiyasi evaziga Java viruslardan va soxtaliklardan holi tizimlarni yaratish imkonini beradi. Autentifikatsiya qilish metodlari ochiq kalitli shifrlashiga asoslangan.

- **Architecture-neutral –** Java kompilyator arxitekturaga bo’g’liq bo’lmagan ob’ektli fayl formatini generatsiya qiladi. Bu esa kompilyatsiya qilingan kodni Java bajarish tizimi mavjud bo’lgan ko’pchilik protsessorlarda bajarilishini ta’minlaydi.
- **Multithreaded –** Javaning ko’p oqimli (multithreaded) funksiyasi yordamida bir vaqtning o’zida bir nechata vazifani (amallarni) bajaruvchi dasturlarni tuzish imkonini beradi.
- **Interpreted –** Java byte code tezkor ravishda qurilma(microprocessor) tiliga tarjima qilinadi va hech qanday joyda saqlanmasdan qurilmaga uzatiladi. **High Performance –** With the use of Just-In-Time compilers, Java enables high performance.
- **Distributed –** Java is designed for the distributed environment of the internet.
- **Dynamic –** Java C yoki C ++ ga qaraganda dinamik deb hisoblanadi, chunki u rivojlanayotgan muhitga moslashishga mo'ljallangan.

JAVA TARIXI

- James Gosling, Mike Sheridan, and Patrick Naughton initiated the Java language project in June 1991. The small team of sun engineers called Green Team.
- Firstly, it was called "Greentalk" by James Gosling and file extension was .gt.
- After that, it was called Oak and was developed as a part of the Green project.
- Java is an island of Indonesia where first coffee was produced (called java coffee).
- Originally developed by James Gosling at Sun Microsystems (which is now a subsidiary of Oracle Corporation) and released in 1995.
- In 1995, Time magazine called Java one of the Ten Best Products of 1995.
- JDK 1.0 released in(January 23, 1996).

There are many java versions that has been released. Current stable release of Java is Java SE 8.

- JDK Alpha and Beta (1995)
- JDK 1.0 (23rd Jan, 1996)
- JDK 1.1 (19th Feb, 1997)
- J2SE 1.2 (8th Dec, 1998)
- J2SE 1.3 (8th May, 2000)
- J2SE 1.4 (6th Feb, 2002)
- J2SE 5.0 (30th Sep, 2004)
- Java SE 6 (11th Dec, 2006)
- Java SE 7 (28th July, 2011)
- Java SE 8 (18th March, 2014)

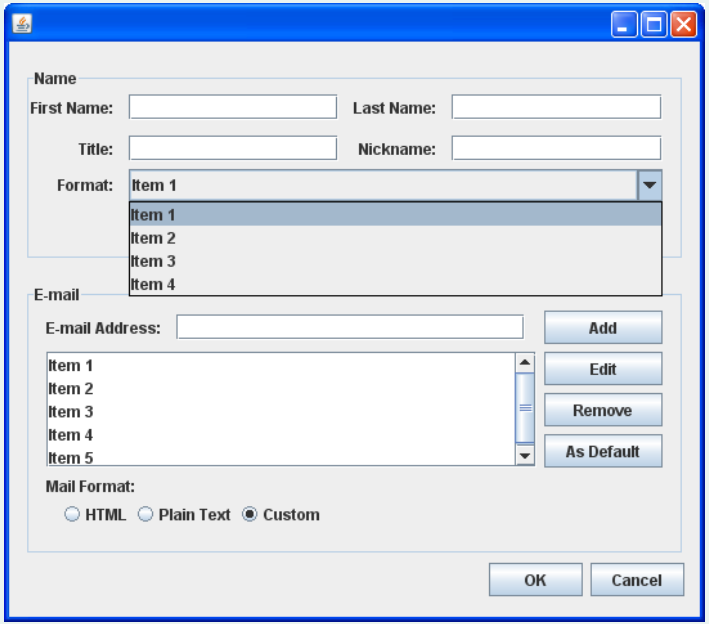
Java-da ishlaydigan ilovalar turlari

Oracle (2010 yilda Sun Microsystems sotib olgan) ma'lumotlariga ko'ra, Java hozirda 3 milliarddan ortiq qurilmalarda ishlaydi.

- **1. Desktop GUI Applications;**
- **2. Mobile Applications;**
- **3. Embedded Systems;**
- **4. Web Applications;**
- **5. Web Servers and Application Servers;**
- **6. Enterprise Applications;**
- **7. Scientific Applications;**

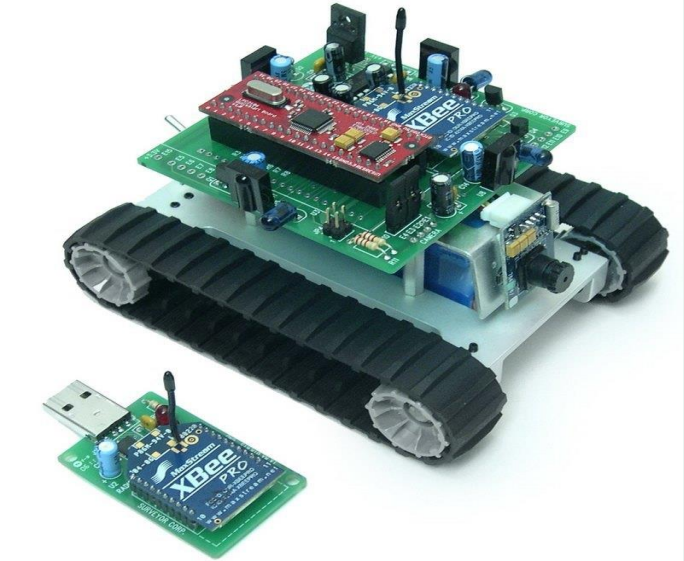
Desktop GUI Applications

- AWT
- Swing
- SWT
- SwingX
- JavaFX
- Apache Pivot
- Qt Jambi



Mobile Applications

Embedded Systems
(Встроенные системы)



Web Applications

PERSONAL
DEVELOPMENT PROCESS
stay weird, stay different

Java Virtual Machine (JVM)

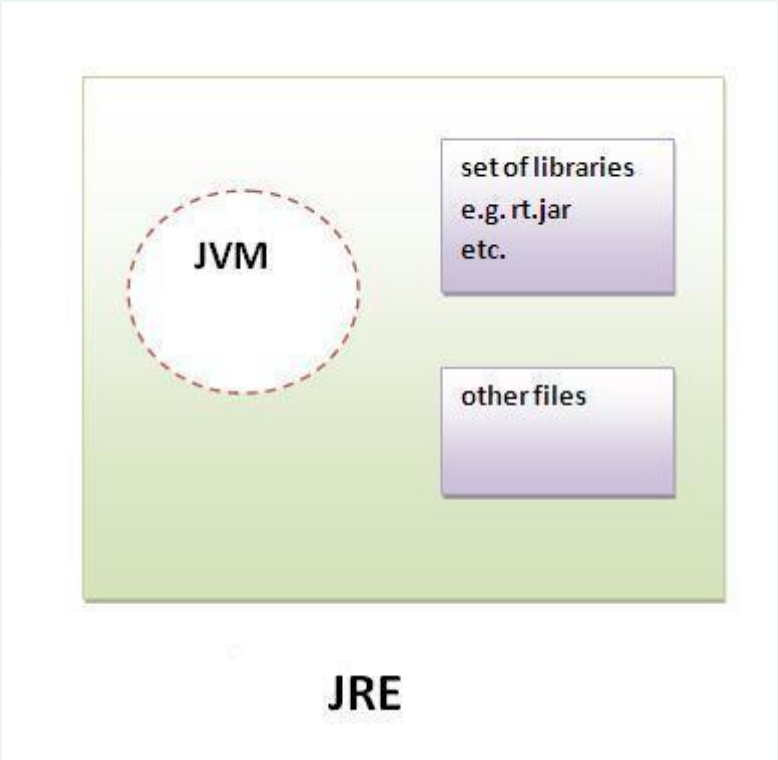
JVM (Java Virtual Machine) is an abstract machine. It is a specification that provides runtime environment in which java bytecode can be executed.

The JVM performs following main tasks:

- **Kodni yuklash (Loads code)**
- **Kodni tekshirish (Verifies code)**
- **Kodni bajarish (Executes code)**
- **Bajarilish muhitini yaratish (Provides runtime environment)**

JAVA RUNTIME ENVIRONMENT (JRE)

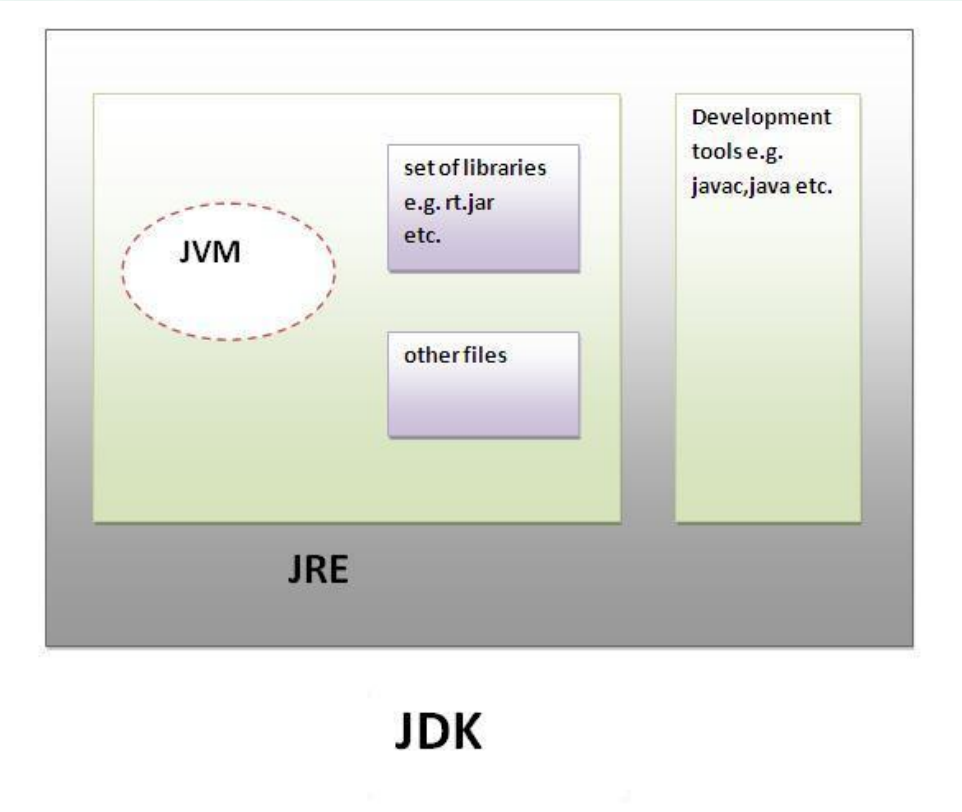
JRE –bajarilish muhitini ta’minlash uchun ishlatiladi. U fizik jihatdan mavjud va uning tarkibida turli kutubxonalar va JVM bajarish jarayonida ishlatiladigan boshqa fayllar mavjud.


JVM
set of libraries
e.g. rt.jar
etc.
other files
JRE

Java Development Kit (JDK)

JDK is an acronym for Java Development Kit. It physically exists. It contains JRE + development tools.

Java Development Kit



Version	Release date	End of Free Public Updates	Extended Support Until
JDK Beta	1995	?	?
JDK 1.0	January 1996	?	?
JDK 1.1	February 1997	?	?
J2SE 1.2	December 1998	?	?
J2SE 1.3	May 2000	?	?
J2SE 1.4	February 2002	October 2008	February 2013
J2SE 5.0	September 2004	November 2009	April 2015
Java SE 6	December 2006	April 2013	December 2018
Java SE 7	July 2011	April 2015	July 2022
Java SE 8 (LTS)	March 2014	**January 2019 for Oracle (commercial)** December 2020 for Oracle (personal use) At least September 2023 for AdoptOpenJDK At least June 2023[8] for Amazon Corretto	December 2030
Java SE 9	September 2017	March 2018 for OpenJDK	N/A
Java SE 10	March 2018	September 2018 for OpenJDK	N/A
Java SE 11 (LTS)	September 2018	At least August 2024[8] for Amazon Corretto September 2022 for AdoptOpenJDK	September 2026
Java SE 12	March 2019	September 2019 for OpenJDK	N/A
Java SE 13	September 2019	March 2020 for OpenJDK	N/A
Java SE 14	March 2020	September 2020 for OpenJDK	N/A
Java SE 15	September 2020	March 2021 for OpenJDK	N/A
Java SE 16	March 2021	September 2021 for OpenJDK	N/A
Java SE 17 (LTS)	September 2021	TBA	TBA

Java Simple example

```
public class MyProgram{
        public static void main(String args[]){
            System.out.println("Hello Java");
        }
}
```

public : access modificator (Dostup yoki ruxsat turi) bo’lib ushbu methodning barchaga ruxsat etilganligini va ko’rinishini ta’minlaydi.

class: kalit so’z bo’lib Javada klasslarnini e’lon qilish uchun ishlatiladi.

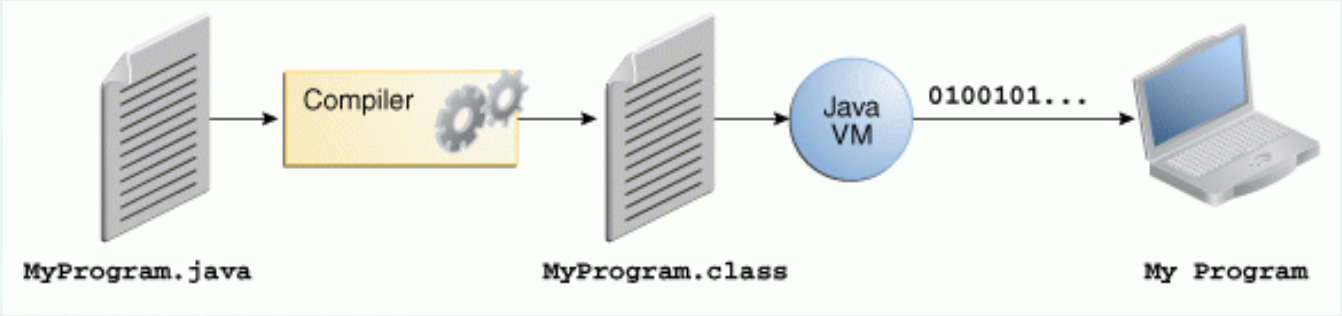
static: Kalit so’z bo’lib methodni static qilsih uchun ishlatilgan. Satatic methodlarga murojat qilish(chaqirish) uchun classdan object olish shart emas. Bu yerda **main()** methodini JVM chaqiradi. Bunda ushbu method joylashgan classdan object olinmaydi.

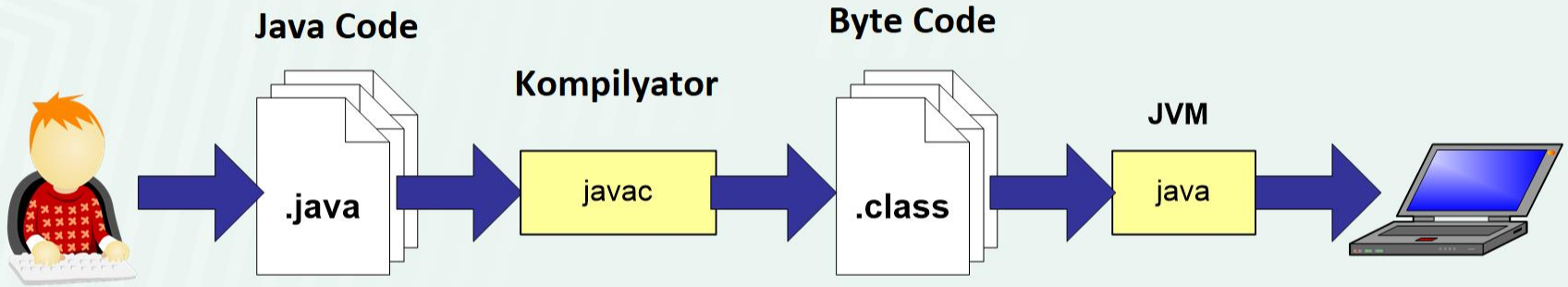
void: qaytarish toifasi bo’lib method hechnima qaytarmasligini bildiradi.

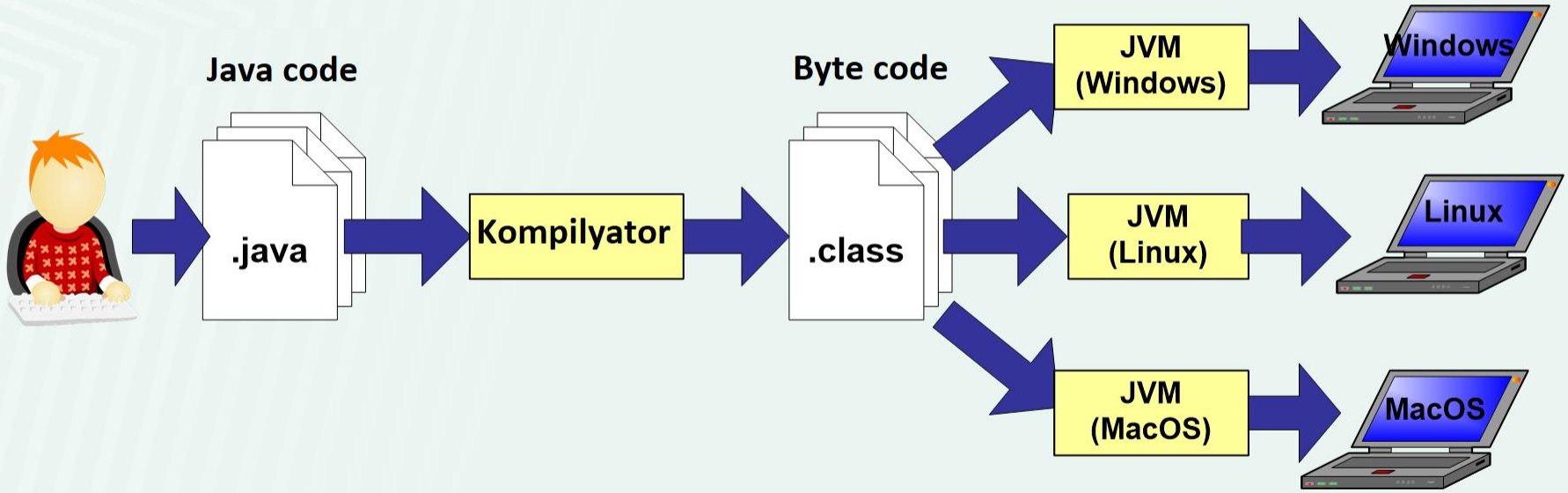
main: main() methodi Javada yozilgan dastur uchun juda ham muhim methoddir. Dasturning barcha logikasi aynan mana shu method ichida bajariladi. Agarda dasturda main() methodi bo’lmasa compilyasiya xatoligi yuz beradi.

String[] args : bu yozuv toifasi String va nomi args bo’lgan massiv(array)ni ifodalaydi. Javada massivlar mavzusida massivlar haqida batafsil to’xtalamiz.

System.out.println : Consolga yozish uchun ishlatiladi.


Compiler
Java VM
0100101...
MyProgram.java
MyProgram.class
My Program


Java Code
Kompilyator
Byte Code
JVM
.java
javac
.class
java


Java code
Byte code
.java
Kompilyator
.class
JVM
(Windows)
JVM
(Linux)
JVM
(MacOS)
Windows
Linux
MacOS

Compile and Run

javac filename.java

java classname

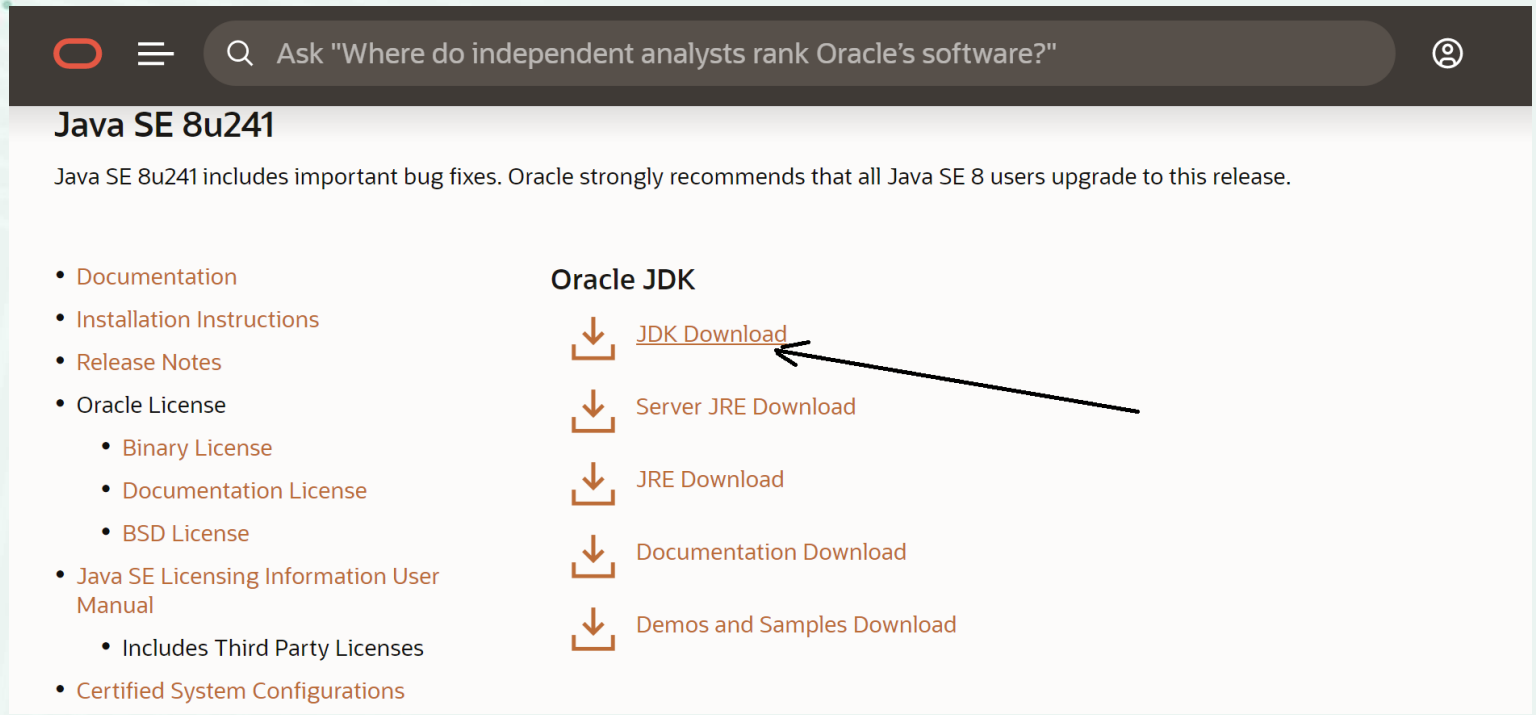
JDK ni o’rnatish

1. JDK o’rnatish paketini yuklash.
2. JDK ni o’natish.
3. O’zgaruvchi muhitini sozlash.
4. Tekshirish.

JDK o’rnatish paketini yuklash.

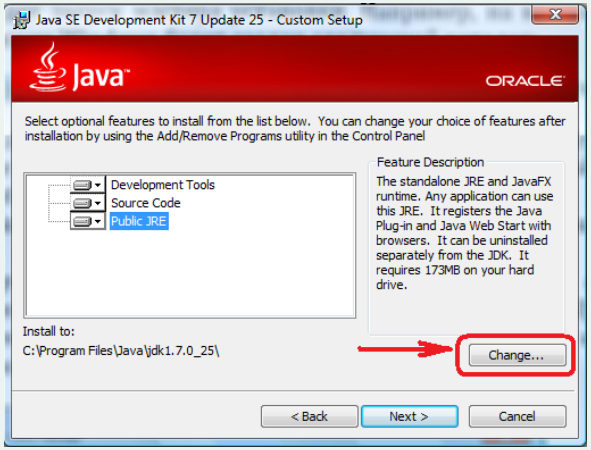
O’rnatish paketini (java 8) quyidagi havola orqli yuklab olish mumkin:

https://www.oracle.com/java/technologies/javase-downloads.html#JDK8


Ask "Where do independent analysts rank Oracle's software?"
Java SE 8u241
Java SE 8u241 includes important bug fixes. Oracle strongly recommends that all Java SE 8 users upgrade to this release.
Documentation
Installation Instructions
Release Notes
Oracle License
Binary License
Documentation License
BSD License
Java SE Licensing Information User Manual
Includes Third Party Licenses
Certified System Configurations
Oracle JDK
JDK Download
Server JRE Download
JRE Download
Documentation Download
Demos and Samples Download

JDK ni o’natish.

Yuklangan faylni ishga tushirish bilan o’rnatish jarayoni boshlanadi.



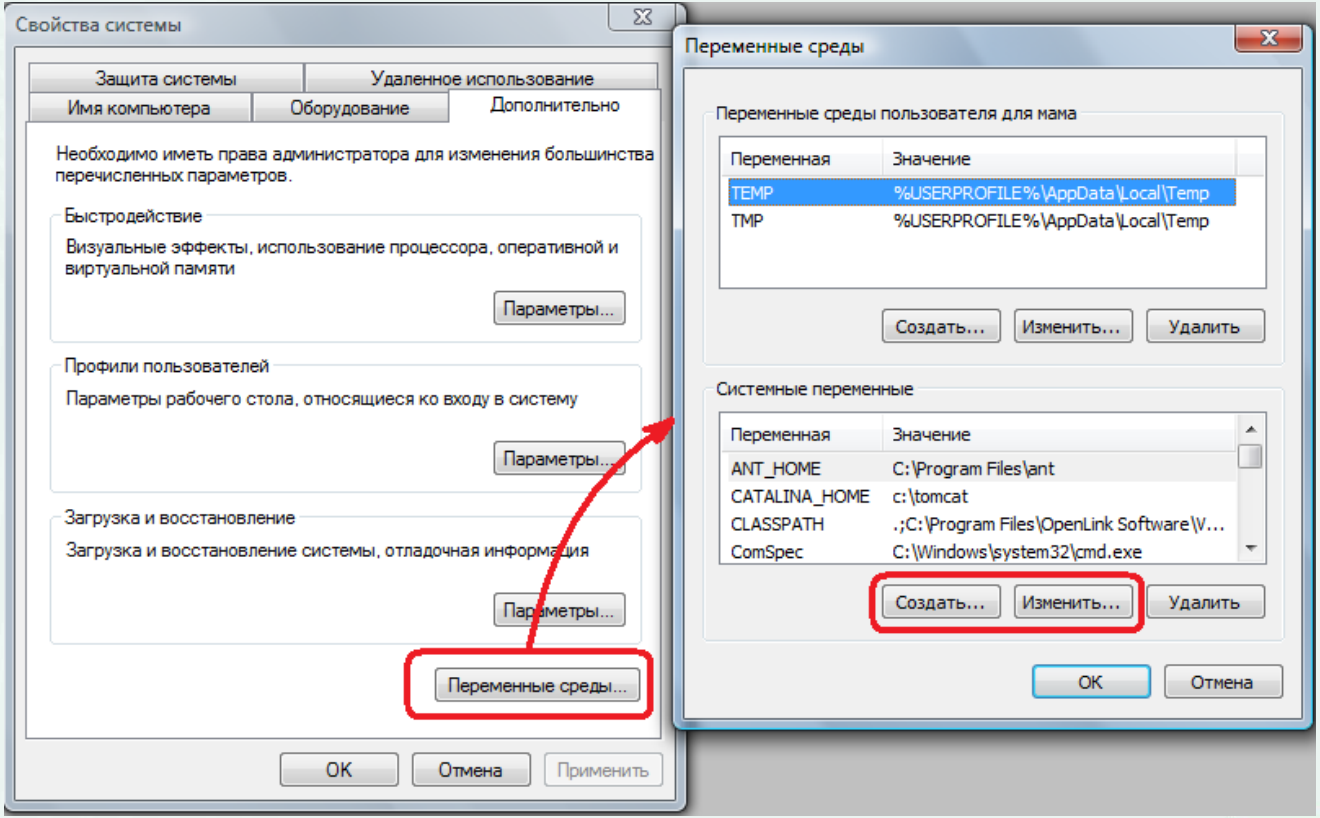
o’rnatish joyini ko’rsatishimiz mumkin. Buning uchun “change” tugmasidan foydalaniladi va bir necha daqiqadan so’ng java o’rnatiladi.

PERSONAL DEVELOPMENT PROCESS
stay weird, stay different

O’zgaruvchi muhitini sozlash.

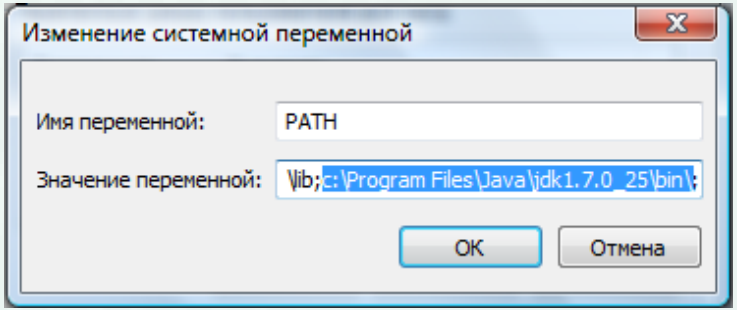
- Keyingi qadamda bajariluvchi fayllarning yo’li ko’rsatiladi, ya’ni sistema o’zgaruvchilari PATH, CLASSPATH va JAVA_HOME.
- Buning uchun **Панель управления -> Система> Дополнительные параметры системы** tanlanadi.

Ochilgan oynadan **Переменные среды** tanlanadi va ekranga chiqqan oynaning **Системные переменные** qismida **PATH** o’zgaruvchi bor-yo’qligi tekshiriladi, agarda o’zgaruvchi yo’q bo’lsa uni yaratish**,** agarda bor bo’lsa uni o’zgartirish talab etiladi



PERSONAL
DEVELOPMENT PROCESS
stay weird, stay different

Keyingi qadamda o’zgaruvchi PATH ning qiymati – Java o’rnatilgan papka yo’li ko’rsatiladi



Bunda agarda o’zgaruvchining qiymati mavjud bo’lsa yangisi ; belgi orqali oxiriga qo’shib qo’yiladi.

Demak bizda ikkita o’zgaruvchi yaratildi yoki o’zgartirildi:

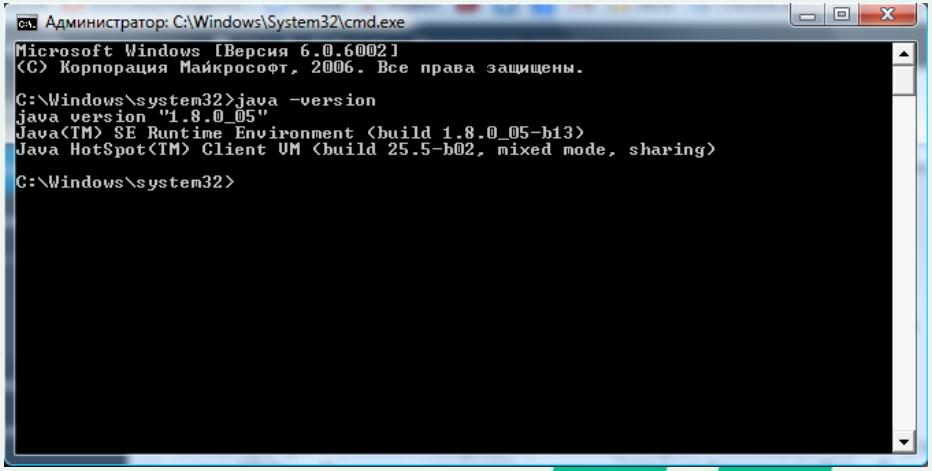
PATH: *C:\Program Files\Java\jdk1.7.0_25\bin*
JAVA_HOME: *C:\Program Files\Java\jdk1.7.0_25*

Tekshirish

O’rnatish muvaffaqiytli kechgan yoki yo’qligini tekshirish uchun buyruqlar satri (командная строка)da

java –version

buyrug’idan foydalaniladi.

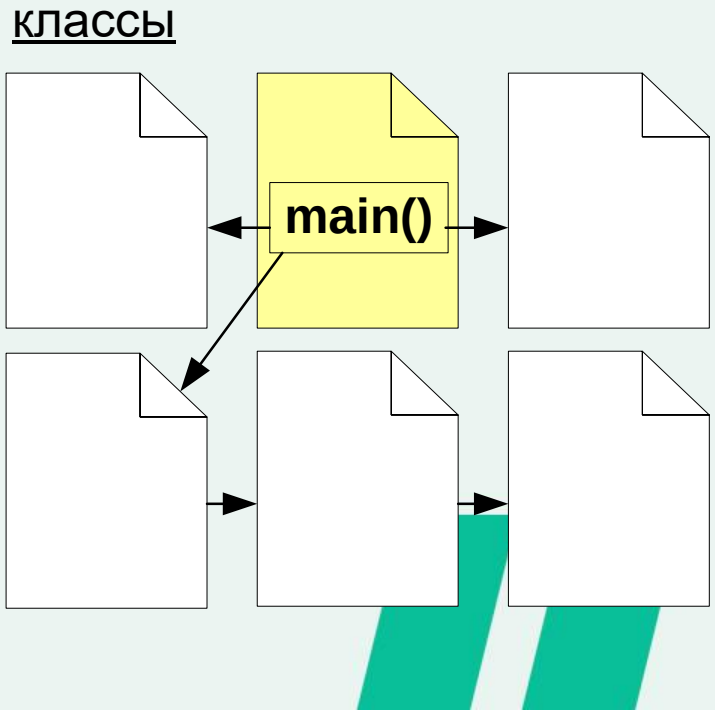


Javada birinchi dastur

Javada dastur ishlashni main() methodidan boshlaydi.

Mathod kirishga kommandalar satrining parametrlar massivini qabul qiladi.

Javada dastur bajarilishi boshlanuvchi klassni asosiy klass deb atash qabul qilingan***(main class***)



PERSONAL
DEVELOPMENT PROCESS
stay weird, stay different

Javada dastur yaratish quyidagi 3 ta bosqichda amalga oshiriladi:

1. Resurs (kod yoziladigan) faylni yaratish (Создание исходного файла)
2. Resurs faylni kompilyatsiya qilish (Компиляция исходного файла)
3. Dasturni ishga tushirish (Запуск программы)

Resurs (kod yoziladigan) faylni yaratish

- Dastlab dastur matnini Javada yozish va saqlash kerak. Bu bizning resurs fayl bo'ladi. Resurs faylni yaratish uchun hohlagan matn muharriridan foydalanishimiz mumkin. Masalan standart NotePad (Блокнот), Notepad++(kod yozish uchun qulayroq), Subline muharrirlari.

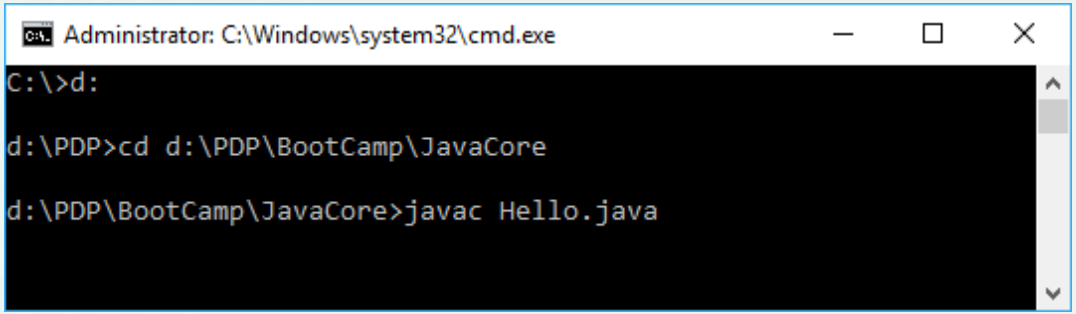
- Kodni yozib bo'lgandan so'ng uni klass nomi bilan bir xil va kengaymasini .java ko'rinishda saqlash kerak.

- ***DIQQAT!*** Fayl nomi klass nomi bilan aniq bir xil bo'lishi (katta, kichik harflarga e'tibor berilsin) va kodirovka ANSI bo'lishi lozim.

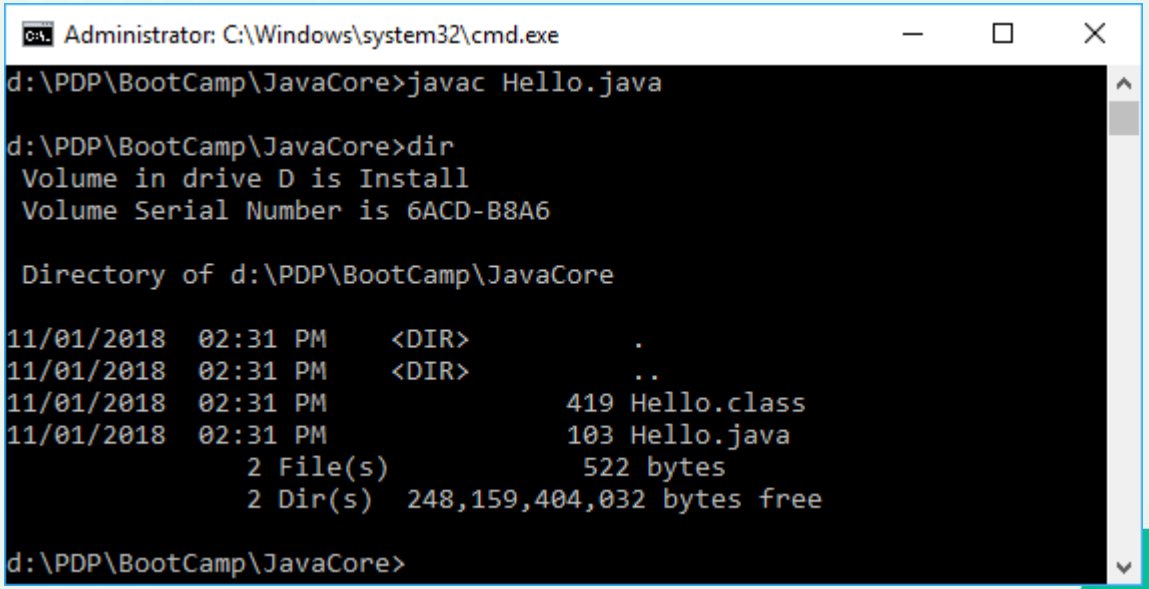
Resurs faylni kompilyatsiya qilish

Javada kompilyatsiya qilish uchun JDK tarkibiga kiruvchi ***javac*** kompilyatori ishlatiladi.

Resurs faylni kompilyatsiya qilish uchun kommandalar satrini ishga tushirib ishchi katalogga o'tiladi va javac resurs_fayl.java yozilib enter bosiladi.

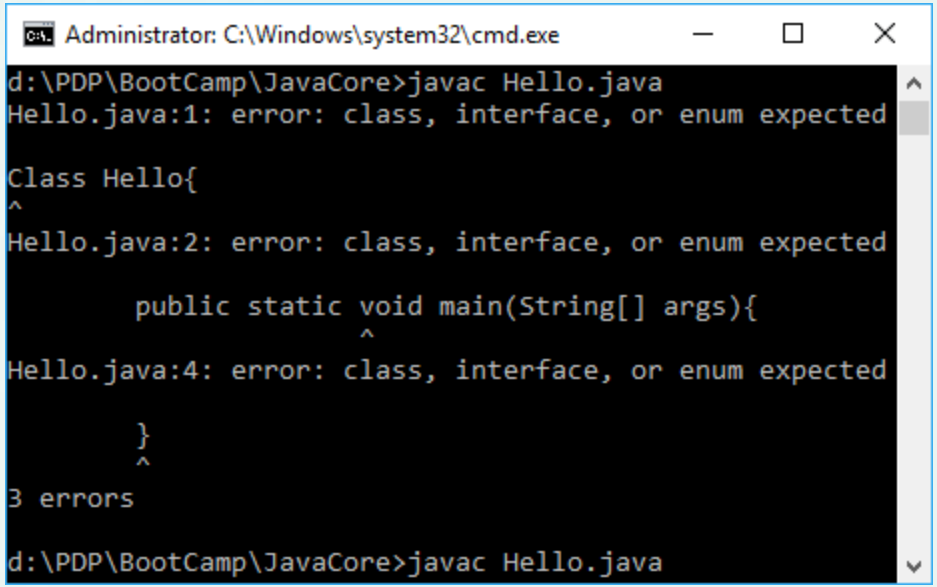


Fayl muvaffaqiyatli kompilyatsiya bo’lsada ekranga buni bildiruvchi hech qanday belgi chiqmaydi. Lekin ishchi katalogda ***resurs_fayl.class*** nomli fayl paydo bo’ladi. Buni ***dir*** buyrug’I orqali tekshirish mumkin:



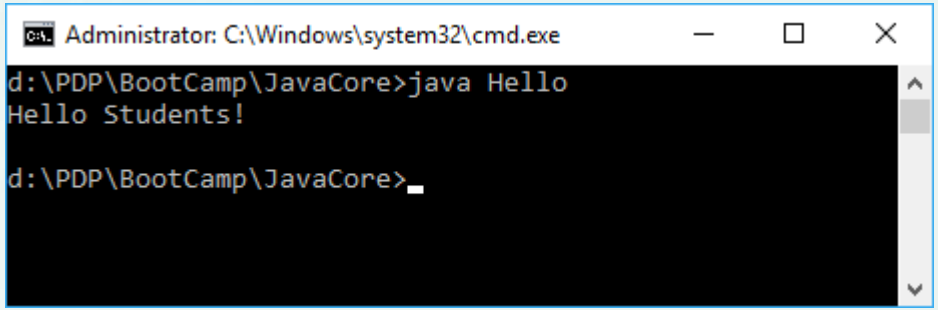
Agarda ro’yxatda ***resurs_fayl.class*** fayli bor bo’lsa kompilyatsiya muvaffaqiyatli amalga oshirilgan, aks holda yo’q.

Agarda dastur kodida xatolik bo’lsa kompilyatsiya bu haqda xabar beradi. Bu holda ***resurs_fayl.java*** faylni ochib xatolikni bartaraf etib fayl saqlanadi va qaytadan kompilyatsiya qilinadi.



Dasturni ishga tushirish

- Komandalar satrida ***java resurs_fayl*** buyrug'I kiritilib enter bosiladi.
- Agarda bundan oldingi amallar to'g'ri bajarilgan bo'lsa dastur natijasi ekranga chiqadi.



Console orqali qiymat kiritish

Javada Console orqali ekrandan qiymat qabul qilish uchun java.util.Scanner kutubxona(class)dan foydalaniladi.

```
import java.util.Scanner;  // java.util paketidan Scanner klassi import qilindi
public class Hi{
	public static void main(String[] args){
		System.out.println("Iltimos ismingizni kiriting: ");
		Scanner scannner = new Scanner(System.in);     //o’zgaruvchini e’lon qilish
		System.out.println("Sizning ismingiz "+scannner.next());
	}
}
```