

Core Java

آموزش ساده و آسان جاوا

به نام خدا

تقدیرم به هموطنان عزیزم

جاوا را با لذت یاد بگیرید!

Core Java

آموزش ساده و آسان جاوا

آموزش زبان برنامه نویسی جاوا

آموزش ساخت پروژه بازی HANGMAN

مدرس : سعید قلندری

جاوا را ساده، آسان و شیرین بنوشید!!!



این جلسه آموزشی رایگان است، فروش و ویرایش آن ممنوع و مرام می باشد. اما این کتاب را می توانید همین جور که هست در سایت و شبکه اجتماعی خود به اشتراک بگذارید.

Core Java

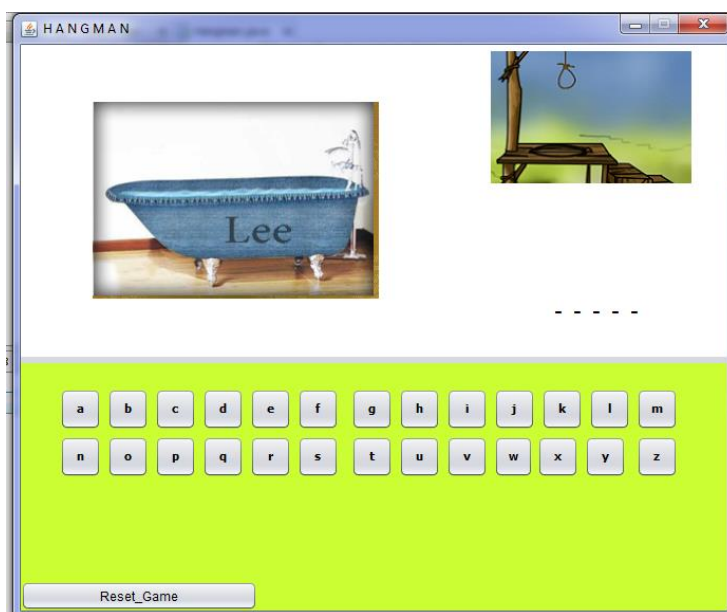
آموزش ساده و آسان جاوا

بسم الله الرحمن الرحيم

در این آموزش قصد داریم شیوه ساختن بازی HANGMAN که سورس کد آن را در سایت جاواپرو به صورت رایگان به اشتراک گذاشتیم بپردازیم.

بازی به این صورت است که یک کلمه به عنوان کلمه رمز در نظر گرفته میشود و کاربر باید این کلمه را حدس بزند، (در اینجا حدس به صورت کل کلمه نیست بلکه حرف به حرف میباشد) در هر بار که حرفی از کلمه رمز اشتباه حدس زده شود، کاربر به یک مرحله از اعدام نزدیک میشود. (تعداد مراحل برای حدس کلمه ۷ تا میباشد. یعنی اگر تعداد حدس به هفت رسید بازی تمام هست و ما بازنده میشویم).

مثل [بازی آفتابه](#) یک تصویر به عنوان کلمه رمز در نظر گرفته میشود و کاربر با کلیک کردن بر روی حروف الفبا (در اینجا حروف انگلیسی در نظر گرفته شده است). حدس خودش را امتحان کند. به تصاویر زیر توجه کنید.



تصویر (۱)

Core Java

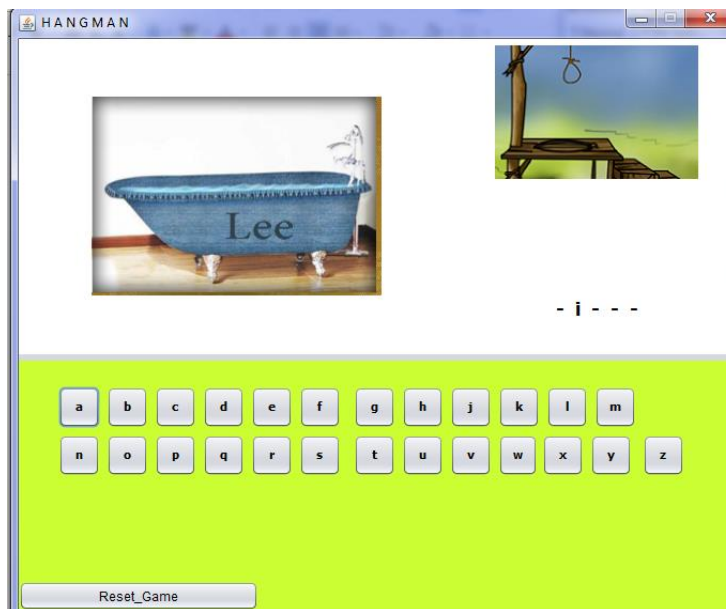
آموزش ساده و آسان جاوا

در تصویر (۱) با زدن دکمه start_Game بازی شروع میشود.

همان طور که در تصویر (۱) مشاهده میکنید ، یک کلمه به صورت خط چین و تصویر همان کلمه در نظر گرفته شده است تعداد خط چین ها به اندازه طول کلمه میباشد.

حال با کلیک کردن بر روی حروف حدس خودمان را امتحان می نماییم اگر حدس ما درست باشد آن حرف در جای خودش قرار میگیرد در غیر این صورت یک مرحله به اعدام نزدیک میشویم. مانند تصاویر (۲) و (۳):

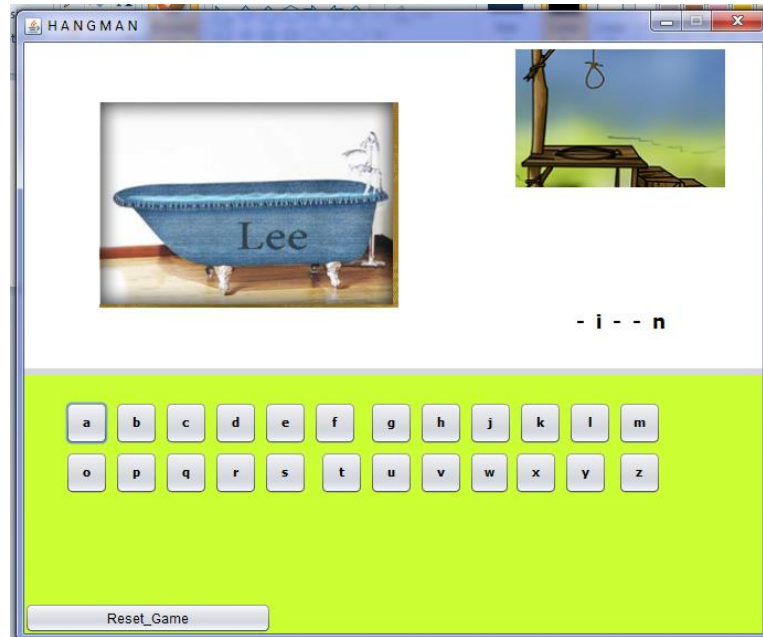
ابتدا حرف **i** و بعد حرف **n** حدس زده ایم که هر دو درست هستند.



تصویر (۲)

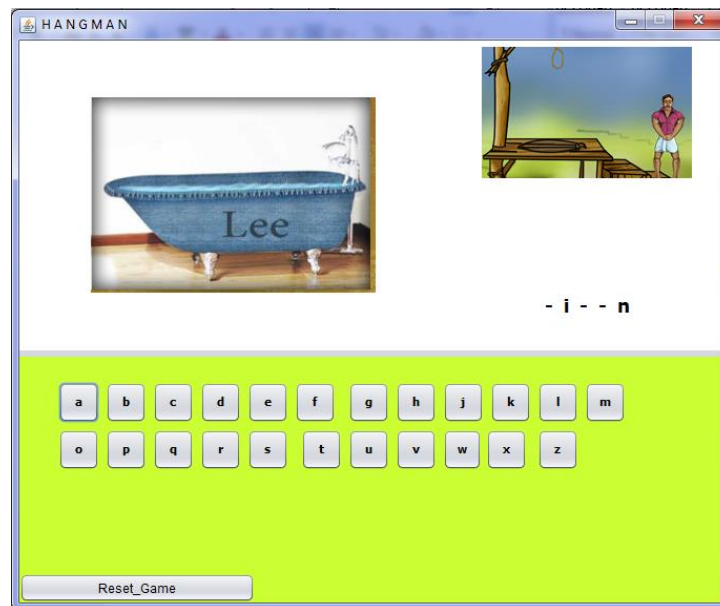
Core Java

آموزش ساده و آسان جاوا



تصویر (۳)

حال در تصویر (۴) یک حرف اشتباه حدس میزنیم:



تصویر (۴)

Core Java

آموزش ساده و آسان جاوا

همانطور که مشاهده میکنید به دلیل اشتباه حدس زدن یک مرحله به اعدام نزدیک میشویم. در ادامه (مثلا میخواهیم بازنده شویم) چگونگی کار نشان داده میشود، به تصاویر زیر دقت نمایید:



تصویر (۵)



تصویر (۶)

Core Java

آموزش ساده و آسان جاوا



تصویر (۷)



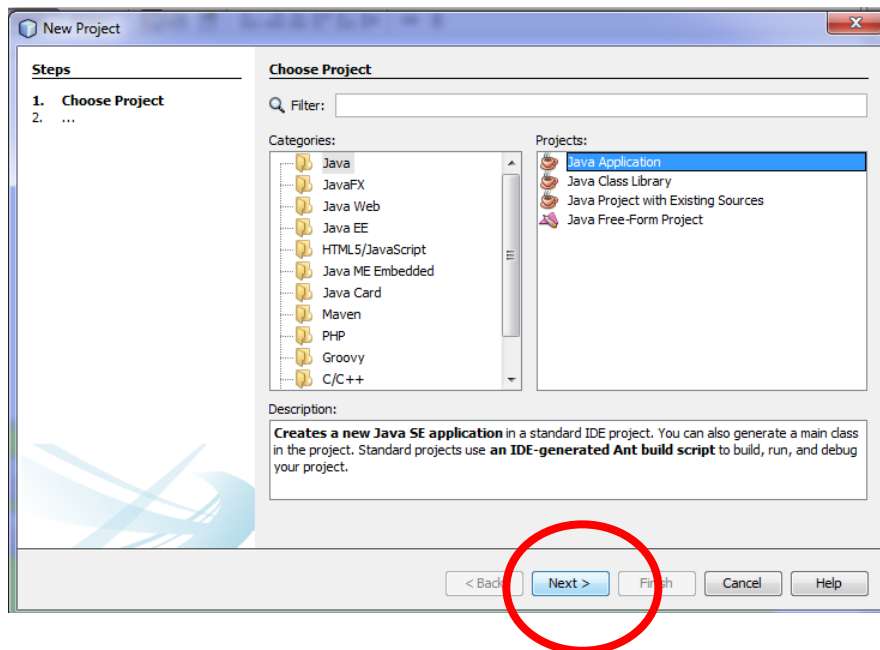
تصویر (۸)

حال قصد داریم این پروژه ساده را پیاده سازی کنیم.

ابتدا یک پروژه جدید ایجاد میکنیم. تصویر (۹)

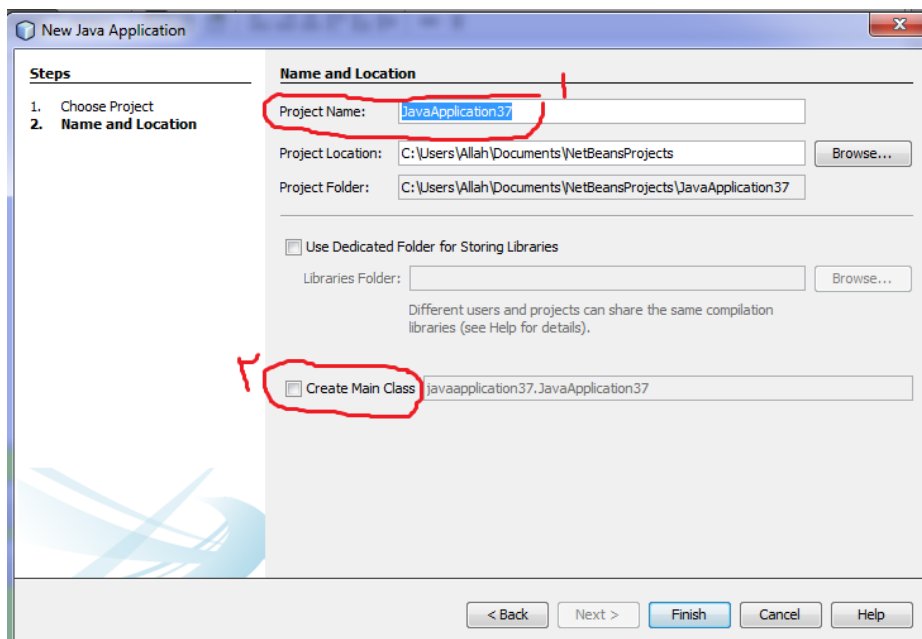
Core Java

آموزش ساده و آسان جاوا



تصویر (۹)

در تصویر (۹)، دکمه Next را زده به مرحله بعد می رویم....



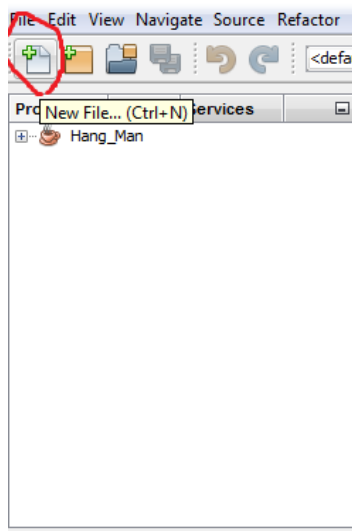
تصویر (۱۰)

Core Java

آموزش ساده و آسان جاوا

- در تصویر (۱۰) در قسمت ۱ اسم دلخواهی برای پروژه خود در نظر میگیریم مثلا Hangman
- در قسمت ۲ تصویر تیک مربوط به mainclass را برمیداریم و دکمه finish را میزنیم.

بعد مطابق تصاویر زیر عمل میکنیم:

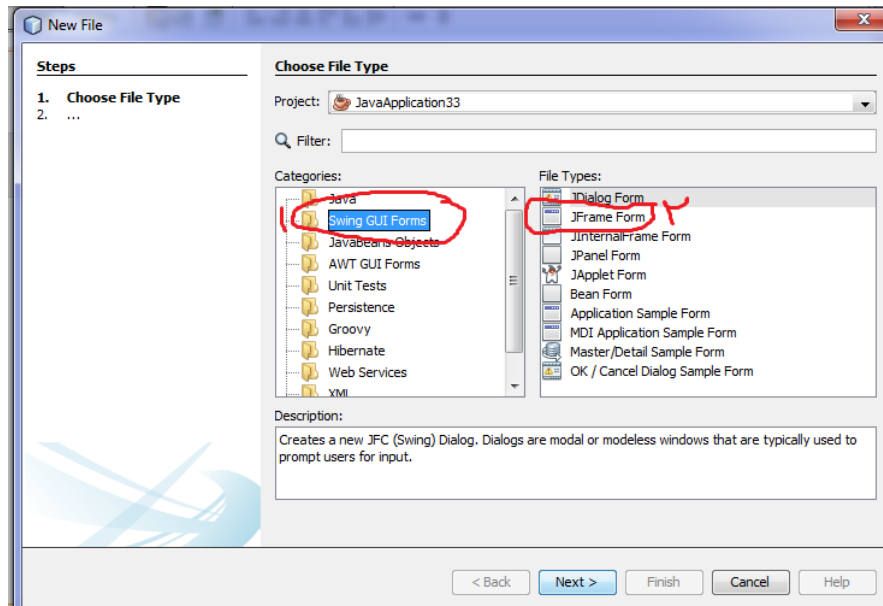


تصویر (۱۱)

در تصویر (۱۱) روی قسمت مشخص شده (New File) کلیک کرده و مرحله زیر را انجام میدهیم:

Core Java

آموزش ساده و آسان جاوا

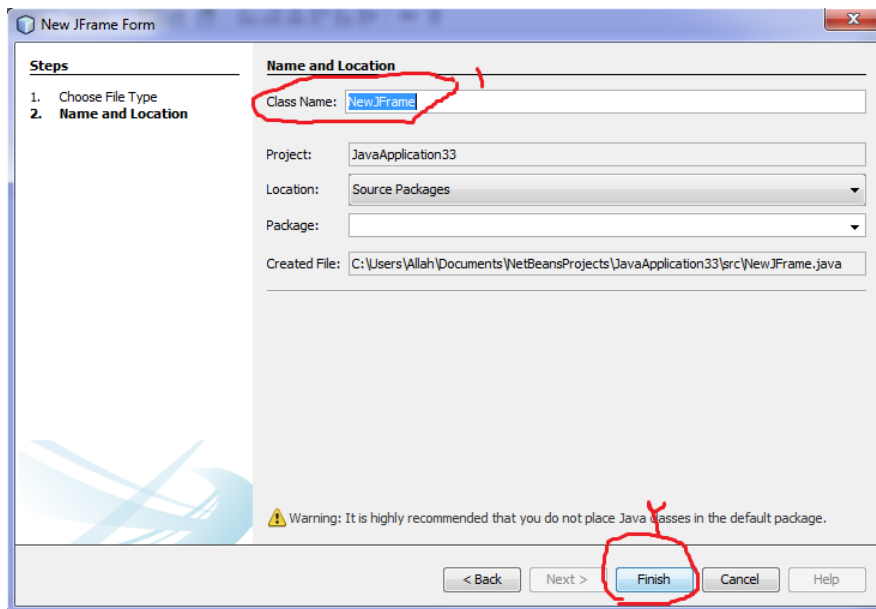


تصویر (۱۲)

در تصویر (۱۲) ابتدا رو قسمت ۱ کلیک کرده و بعد قسمت ۲ را انتخاب میکنیم (JFrame Form) و دکمه Next را میزنیم.

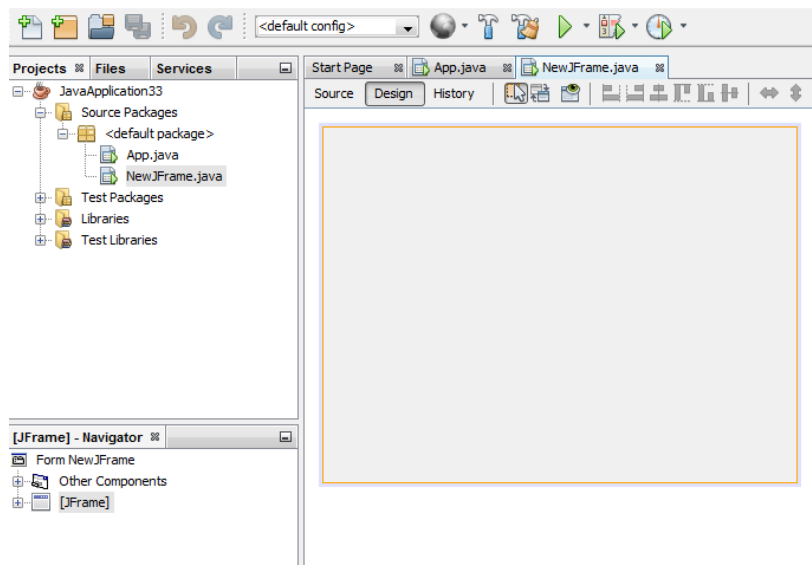
Core Java

آموزش ساده و آسان جاوا



تصویر (۱۳)

در تصویر (۱۳)، در قسمت ۱ تصویر یک نام دلخواه مثل اسم کلاس ها در نظر گرفته و با زدن دکمه finish در قسمت ۲ تصویر کار را اتمام میکنیم و قسمت Form ما آماده میشود مانند تصویر (۱۴):



تصویر (۱۴)

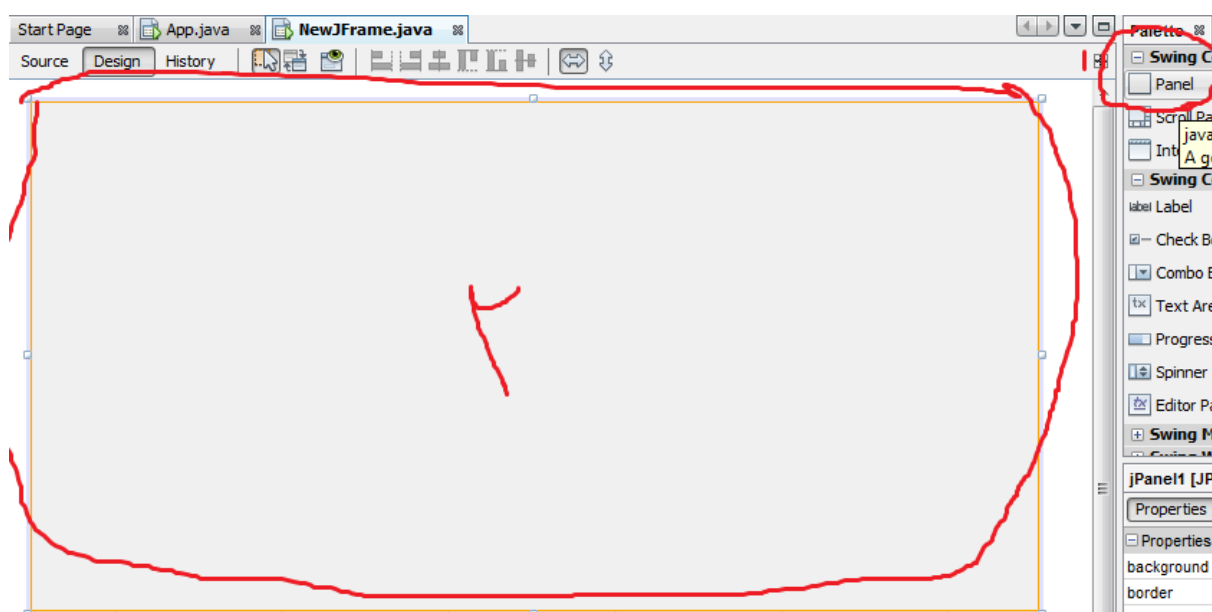
Core Java

آموزش ساده و آسان جاوا

در این بخش ظاهر پروژه خود را آماده میکنیم:

ابتدا مطابق تصاویر (۱۵) و (۱۶) یک Panel انتخاب کرده (قسمت ۱) و آن را بروی Fram قرار میدهیم و آن را قدری میکشیم تا کل عرض form خود را گرفته و تا تقریباً نصف ارتفاع Fram قرار گیرد (قسمت ۲)

(این طراحی دلخواهی است شما میتوانید بهتر از اینجانب طراحی نمایید.)



تصویر (۱۵)

Core Java

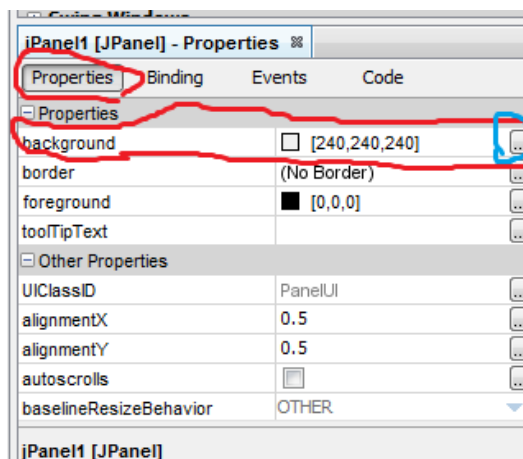
آموزش ساده و آسان جاوا



تصویر (۱۶)

مطابق تصاویر (۱۷) و (۱۸) :

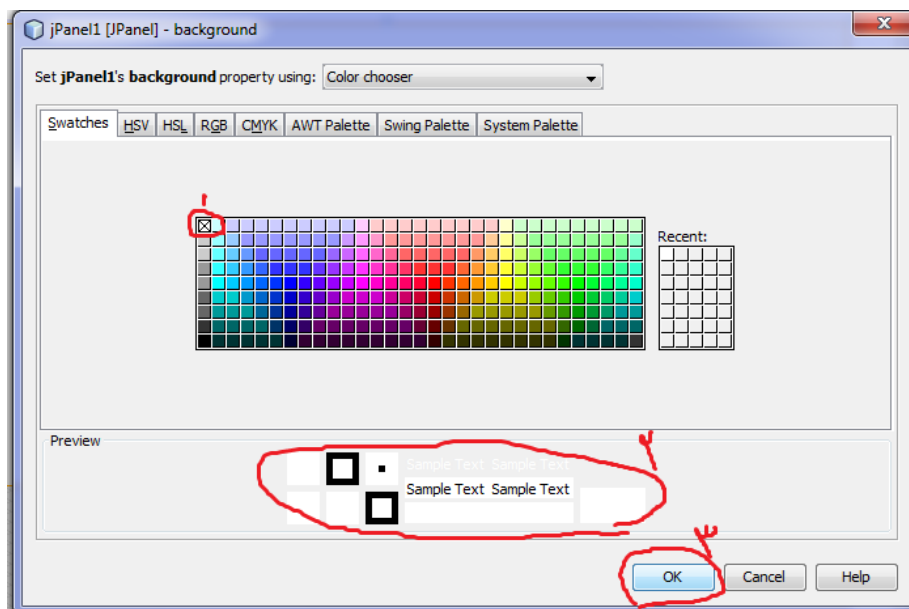
در بخش properties مربوط به Panel قسمت مربوط به background انتخاب نموده و رنگ سفید را انتخاب مینماییم



تصویر (۱۷)

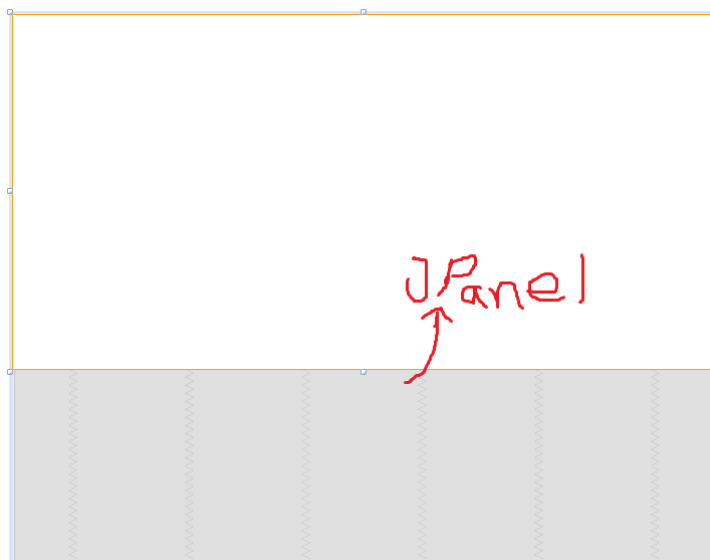
Core Java

آموزش ساده و آسان جاوا



تصویر (۱۸)

همانطور که در تصویر (۱۹) مشاهده میکنید رنگ مربوط به JPanel سفید شده است:



تصویر (۱۹)

Core Java

آموزش ساده و آسان جاوا

در قسمت JPanel طراحی زیر را انجام می‌دهیم. به تصویر (۲۰) دقت کنید:



تصویر (۲۰)

توضیحات مربوط به تصویر (۲۰):

همانطور که در تصویر (۲۰) مشاهده می‌کنید از قسمت ۱ چهار عدد JLabel انتخاب می‌کنیم

جLabel1: (قسمت ۲ تصویر) برای عکس مربوط به کلمه می‌باشد.

Core Java

آموزش ساده و آسان جاوا

JLabel2: (قسمت ۳ تصویر) برای عکس مربوط به چوبه دار میباشد.

JLabel3: (قسمت ۴ تصویر) برای نوشته مربوط به اینکه ما برنده شدیم یا بازنده .

JLabel4: (قسمت ۵ تصویر) برای نوشته مربوط به کلمه رمز مطابق تصویر میباشد.

تصویر (۲۱) را ببینید:

پروژه به صورت زیر می باشد.



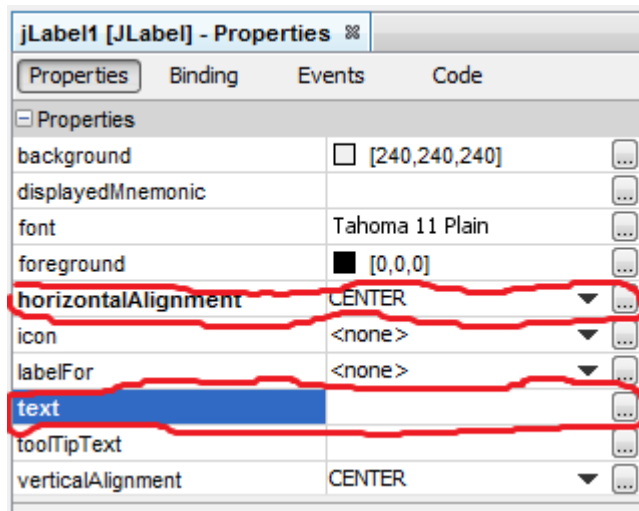
تصویر (۲۱)

- دلیل اینکه از JLabel استفاده میکنیم این است که JLabel ویژگی برای عکس تحت عنوان icon دارد و هم برای اینکه بخواهیم رشته ای (String) بنویسیم.

بعد از اینکه JLabel ها را انتخاب کردیم طبق تصویر (۲۲) برای هر ۴ عدد JLabel کارهای زیر را اعمال میکنیم:

Core Java

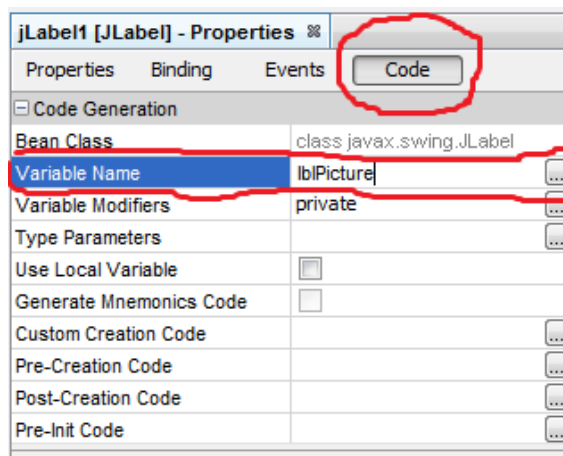
آموزش ساده و آسان جاوا



تصویر (۲۲)

Text را پاک میکنیم و horizontalAlignment را برابر CENTER قرار میدهیم باعث میشود تصویر وسط JLabel قرار گیرد.

بعد مثل تصویر (۲۳) داریم:



تصویر (۲۳)

Core Java

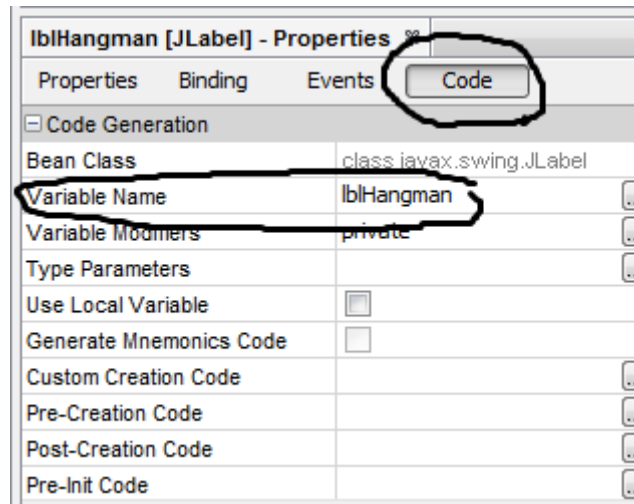
آموزش ساده و آسان جاوا

در تصویر (۲۳)، در قسمت Code قسمت Variable Name یک نام دلخواهی قرار میدهیم. این نام باعث میشود در بخش کد نویسی شناخته شود (در جلوتر به آن می پردازیم)

برای سه JLabel دیگر همین کارها را انجام میدهیم:

توجه داشته باشید در قسمت Code هر JLabel قسمت Variable Name باید اسم های متفاوتی داشته باشد. یعنی هر JLabel یا هر کدام از اجزای گرافیکی دیگر مثل دکمه ها و ... باید نام های متفاوتی داشته باشند.

برای قسمت های مربوط به چوبه دار و وضعیت برد یا باخت و قسمت کلمه رمز اسمهای متفاوتی در نظر میگیریم به تصاویر زیر دقت کنید:



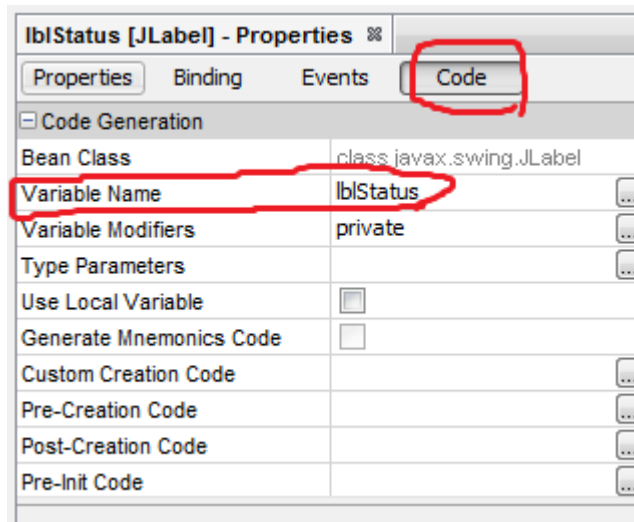
تصویر (۲۴)

تصویر (۲۴) برای چوبه دار است.

تصویر زیر برای قسمت برنده یا بازنده می باشد که ما اسم آن را status گذاشته ایم.

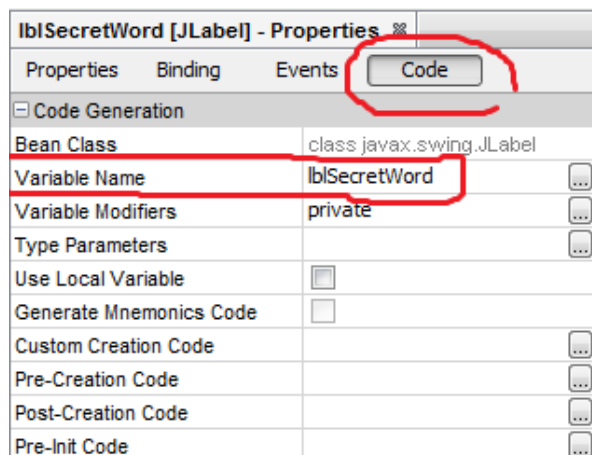
Core Java

آموزش ساده و آسان جاوا



تصویر (۲۵)

تصویر (۲۶) نیز برای کلمه رمز می باشد:



تصویر (۲۶)

خب تا اینجا کار با طراحی مربوط به تصاویر آشنا شدیم حال به طراحی مربوط به دکمه ها می پردازیم.

به تصویر (۲۶) بار دیگر نگاه کنید:

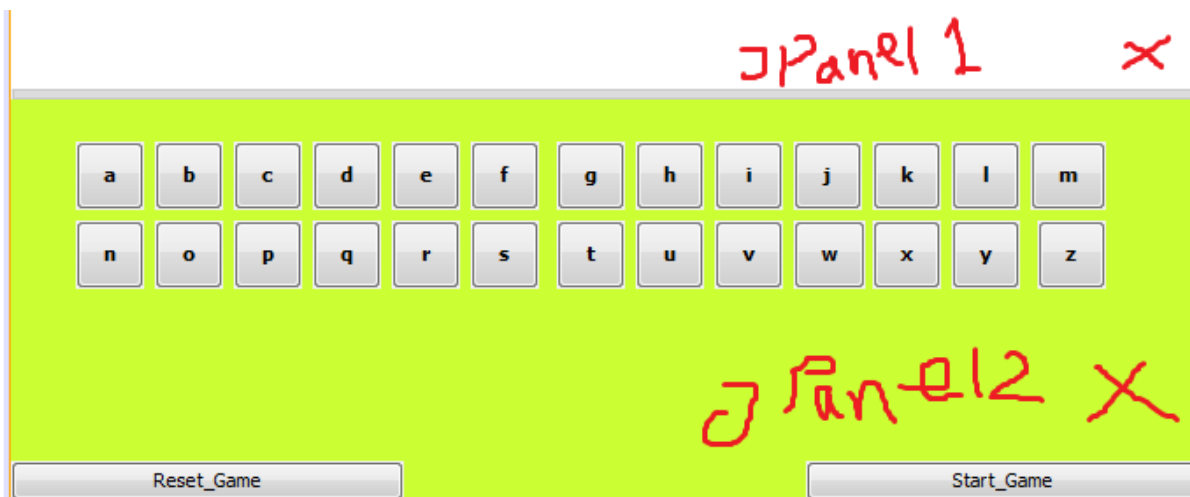
Core Java

آموزش ساده و آسان جاوا



تصویر (۲۷)

قسمت JPanel مربوط به تصاویر بود که طراحی آن تمام شد. حال در قسمت پایین JPanel یک دیگری قرار میدهیم و رنگ background آن را نیز تغییر میدهیم و دکمه ها را در این قسمت میگذاریم:



تصویر (۲۸)

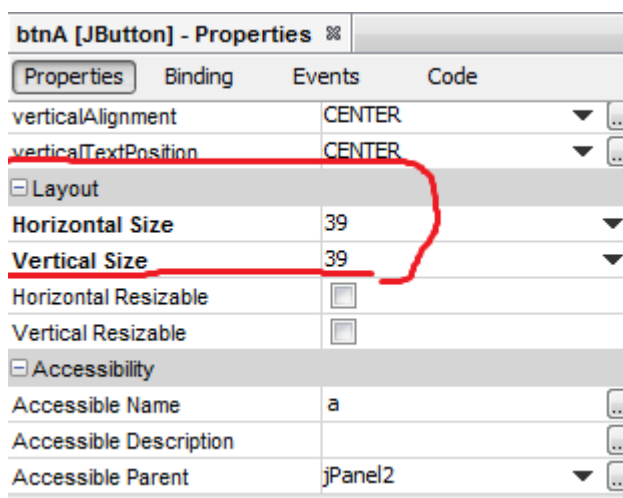
Core Java

آموزش ساده و آسان جاوا

در تصویر (۲۷) jpanel2 که با رنگ سبز مشخص شده محل قرارگیری دکمه ها می باشد. به تعداد حروف الفبا دکمه قرار می دهیم. برای هر کدام از دکمه های الفبا کارهای زیر را انجام می دهیم:



تصویر (۲۹)

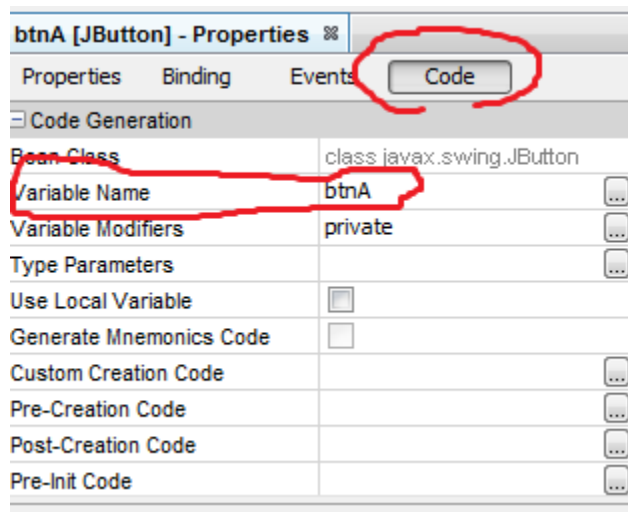


تصویر (۳۰)

Core Java

آموزش ساده و آسان جاوا

تصویر (۲۹) اندازه دکمه را تعیین کرده ایم باری تمام دکمه های حروف این اندازه در نظر گرفته شده شما میتوانید اندازه های دیگری هم در نظر بگیرید.



تصویر (۳۱)

تصویر (۳۰) برای code دکمه a است که ما اسم آن را btnA قرار داده ایم.

حال برای سایر حروف همین کارها را میکنیم و در قسمت code ، variable Name را متناسب با هر حرفی که بود قرار میدهیم.

b-----→btnB

c-----→btnC

...

Core Java

آموزش ساده و آسان جاوا



تصویر (۳۲)

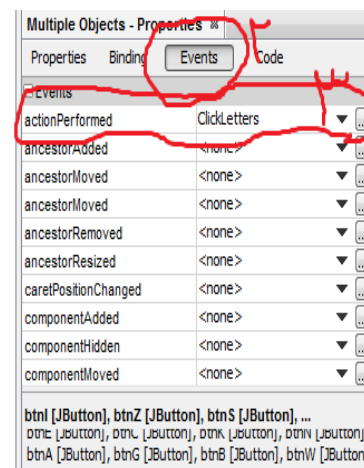
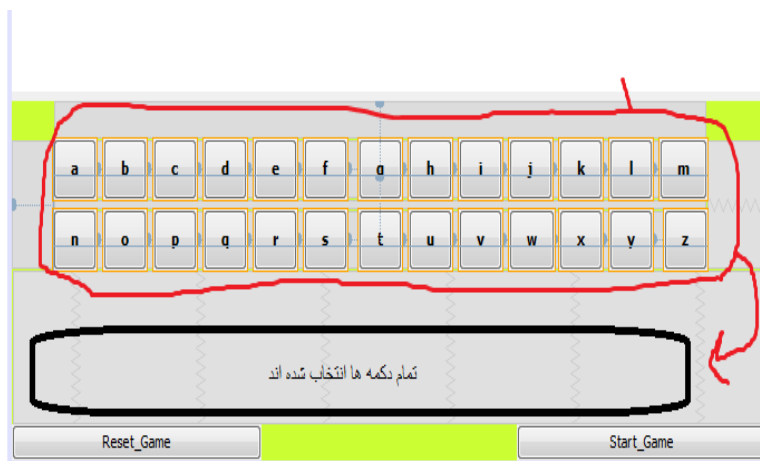
بعد از دکمه های مربوط به حروف دو دکمه یکی برای شروع بازی و دیگری برای reset کردن بازی قرار می دهیم. طراحی تمام شد حال به سراغ کدنویسی آن میرویم.

از آنجا که قرار است با کلیک کردن بروی حروف، آن حرف کلیک شده مقدارش در کلمه قرار گیرد باید رویداد مربوط به کلیک کردن را فعال کنیم.

تمام دکمه های حروف را انتخاب می کنیم (با تک کلیک و نگه داشتن (Ctrl) و event مربوطه را فعال می کنیم: تصویر (۳۲)

Core Java

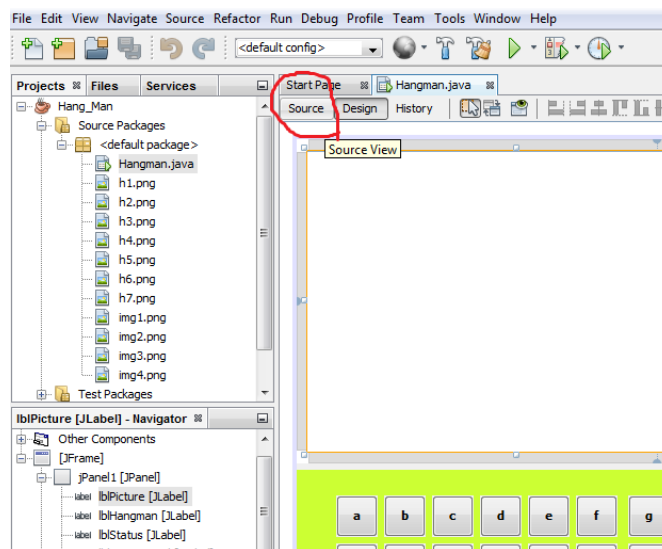
آموزش ساده و آسان جاوا



تصویر (۳۳)

مطابق تصویر (۳۲) در قسمت ۱ تمامی دکمه های حروف انتخاب شده اند در قسمت ۲ وارد Events می شویم در قسمت ۳ اسم دلخواهی برای events انتخاب میکنیم ما اسم ان را ClickLetters گذاشته ایم. بعد از این کار این عمل در قسمت source برنامه فعال میشود.

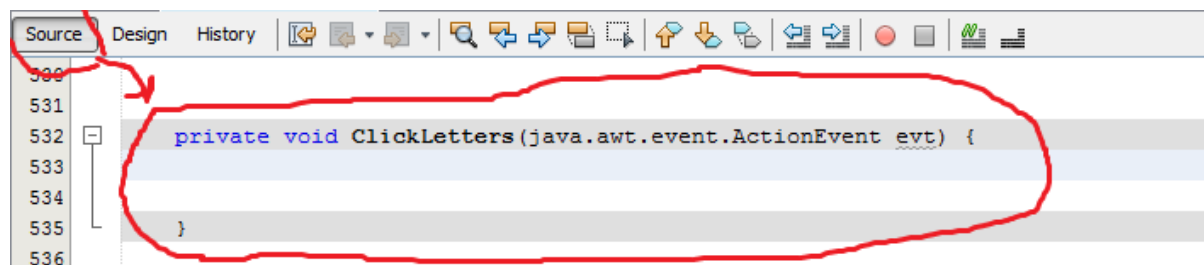
تصاویر زیر:



تصویر (۳۴)

Core Java

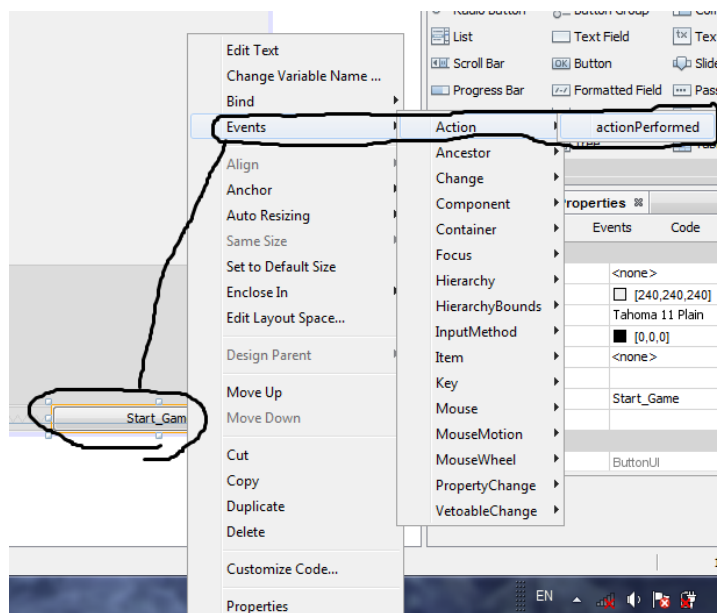
آموزش ساده و آسان جاوا



تصویر (۳۵)

متدی که در تصویر (۳۴) مشاهده میکنید جایی است که با هر بار کلیک کردن بروی دکمه های حروف محتوای آن حرف نمایش داده میشود.

حال مطابق تصاویر زیر برای هر دو دکمه start_Game و Reset_Game events را فعال میکنیم: با کلیک راست کردن بر روی هر دو دکمه و مسیر زیر را انتخاب کردن.



تصویر (۳۶)

Core Java

آموزش ساده و آسان جاوا

بعد از این کار دو متد زیر در قسمت سورس برنامه فعال میشود:

```
private void btnResetGameActionPerformed(java.awt.event.ActionEvent evt) {
    |
}
```

تصویر(۳۷)

```
private void btnStartGameActionPerformed(java.awt.event.ActionEvent evt) {
    |
}
```

تصویر(۳۸)

الگوریتم برنامه:

برنامه به چه صورت انجام میشود.

به این صورت که با کلیک کردن بر روی دکمه Start_Game وارد بازی میشویم بعد به صورت تصادفی یک کلمه که درون آرایه ای قرار داده ایم انتخاب میشود و تصویر مربوط به آن کلمه نیز فعال میشود بعد به تعداد حروف کلمه خط چین در قسمت مربوط به کلمه رمز فعال میشود.

حال کاربر با کلیک بروی حروف حدس خود را امتحان میکند اگر درست بود آن حرف در محل خودش قرار میگیرد در غیر این صورت چوبه دار فعال میشود.

این کار تا ۷ حدس ادامه میابد اگر تا این تعداد حدس کلمه را پیدا کردیم که برنده میشویم در غیر این صورت بازنده هستیم.

مراحل کد نویسی:

ابتدا طبق تصویر(۳۸) تعدادی متغیر را تعریف میکنیم:

Core Java

آموزش ساده و آسان جاوا

```

//In the name of Allah
import javax.swing.*;
import java.util.*;

public class Hangman extends javax.swing.JFrame {
    ImageIcon [] imgHangman ;
    ImageIcon [] imgWord;
    JButton [] buttons;

    String [] secret_word;
    char [] secret;
    char [] blanks;

    Random r;
    int x;

    String secWord;
    String dash;
    String help;
    int cnt;
    boolean flag;

    String win;
  
```

Diagram illustrating the code structure with numbered annotations (1-11) pointing to various elements:

- 1: `import javax.swing.*;`
- 2: `ImageIcon [] imgHangman ;`
- 3: `ImageIcon [] imgWord;`
- 4: `JButton [] buttons;`
- 5: `String [] secret_word;`
- 6: `char [] secret;`
- 7: `char [] blanks;`
- 8: `Random r;`
- 9: `int x;`
- 10: `String secWord;`
- 11: `String dash;`
- 12: `String help;`
- 13: `int cnt;`
- 14: `boolean flag;`
- 15: `String win;`

تصویر (۳۹)

بررسی قسمتهای تعیین شده در تصویر (۳۹):

قسمت ۱:

کتابخانه swing برای کار کردن با اجزای گرافیکی import میکنیم.

کتابخانه util برای استفاده از کلاس Random import میکنیم.

قسمت ۲:

سه آرایه یک بعدی تعریف کرده ایم.

آرایه `imgHangman`: از نوع `ImageIcon` میباشد که قرار است تصاویر مربوط به چوبه دار درونش ذخیره شود.

آرایه `imgWord`: از نوع `ImageIcon` میباشد که قرار است تصاویر مربوط به کلمات رمز درونش ذخیره شود.

Core Java

آموزش ساده و آسان جاوا

آرایه buttons : از نوع JButton بوده و ۲۶ حرف الفبا درونش ذخیره میکنیم.

قسمت ۳:

یک آرایه از نوع رشته ای تحت عنوان secret_word تعریف کرده ایم. درون این آرایه لغاتی تحت عنوان کلمه های رمز ذخیره میکنیم.

قسمت ۴:

دو آرایه از نوع کارکتری تعریف کرده ایم.

آرایه secret که از نوع کارکتر است بعد از اینکه یک کلمه به صورت تصادفی انتخاب شد به صورت آرایه ای از کارکترها تبدیل میکنیم و درون این آرایه ذخیره میکنیم.

آرایه blanks که از نوع کارکتر است به تعداد کارکترهای درون آرایه secret خط چین(-) قرار میدهیم.

قسمت ۵ :

از کلاس Random یک شی ایجاد میکنیم

و یک متغیر از نوع عدد صحیح به نام X تعریف کرده ایم

در قسمت جلوتر با نحوه کار با این دو متغیر آشنا میشویم.

قسمت ۶ و ۷ و ۸ :

سه متغیر از نوع رشته تعریف کرده ایم.

secWord : کلمه رمز انتخاب شده درونش قرار میگیرد

dash : برای بار اول به تعدا حروف کلمه رمز خط چین قرار میگیرد

Core Java

آموزش ساده و آسان جاوا

help : رشته ای است که هر بار حرف صحیح حدس زده میشود با مقادیر آرایه blanks جمع میشود و درونش ذخیره میشود.
قسمت ۹ :

یک عدد صحیح به عنوان cnt در نظر گرفته میشود که تعداد دفعات حدس اشتباه درونش ذخیره میشود.

قسمت ۱۰ :

متغیری از نوع Boolean است که درست یا اشتباه حدس زدن ما را تعیین میکند.

قسمت ۱۱ :

این متغیر رشته ای برای حالت برنده شدن تعیین شده است.

با تمامی این قسمت ها در کد نویسی آشنا میشوید.

قدم بعدی باید درون سازنده کلاس برخی از این متغیرها مقداردهی شوند:

طبق تصاویر زیر:

Core Java

آموزش ساده و آسان جاوا

```
public Hangman() {
    initComponents();

    imgHangman = new ImageIcon[7];
    imgHangman[0]= new ImageIcon(this.getClass().getResource("/h1.png"));
    imgHangman[1]=new ImageIcon(this.getClass().getResource("/h2.png"));
    imgHangman[2]=new ImageIcon(this.getClass().getResource("/h3.png"));
    imgHangman[3]=new ImageIcon(this.getClass().getResource("/h4.png"));
    imgHangman[4]=new ImageIcon(this.getClass().getResource("/h5.png"));
    imgHangman[5]=new ImageIcon(this.getClass().getResource("/h6.png"));
    imgHangman[6]=new ImageIcon(this.getClass().getResource("/h7.png"));

    secret_word = new String[4];
    secret_word[0]="setar";
    secret_word[1]="domino";
    secret_word[2]="livan";
    secret_word[3]="neyrang";

    imgWord = new ImageIcon[4];
    imgWord[0]= new ImageIcon(this.getClass().getResource("/img1.png"));
    imgWord[1]= new ImageIcon(this.getClass().getResource("/img2.png"));
    imgWord[2]= new ImageIcon(this.getClass().getResource("/img3.png"));
    imgWord[3]= new ImageIcon(this.getClass().getResource("/img4.png"));
}
```

تصویر (۴۰)

Core Java

آموزش ساده و آسان جاوا

```
buttons = new JButton[26];
buttons[0]=btnA;
buttons[1]=btnB;
buttons[2]=btnC;
buttons[3]=btnD;
buttons[4]=btnE;
buttons[5]=btnF;
buttons[6]=btnG;
buttons[7]=btnH;
buttons[8]=btnI;
buttons[9]=btnJ;
buttons[10]=btnK;
buttons[11]=btnL;
buttons[12]=btnM;
buttons[13]=btnN;
buttons[14]=btnO;
buttons[15]=btnP;
buttons[16]=btnQ;
buttons[17]=btnR;
buttons[18]=btnS;
buttons[19]=btnT;
buttons[20]=btnU;
buttons[21]=btnV;
buttons[22]=btnW;
buttons[23]=btnX;
buttons[24]=btnY;
buttons[25]=btnZ;
```

تصویر (۴۱)

Core Java

آموزش ساده و آسان جاوا

```
buttons[21]=btnV;  
buttons[22]=btnW;  
buttons[23]=btnX;  
buttons[24]=btnY;  
buttons[25]=btnZ;  
  
for(JButton bt : buttons)  
    bt.setVisible(false);  
  
btnResetGame.setVisible(false);  
}
```

تصویر (۴۲)

قدم بعد این است که متد مربوط به کلیک کردن برای شروع بازی پیاده سازی کنیم:

Core Java

آموزش ساده و آسان جاوا

```
private void btnStartGameActionPerformed(java.awt.event.ActionEvent evt) {  
    r= new Random();  
    x=r.nextInt(secret_word.length);  
    secWord="";  
    secWord=secret_word[x];  
  
    lblPicture.setIcon(imgWord[x]);  
    secret=new char[secWord.length()];  
    blanks= new char[secWord.length()];  
    dash="";  
    cnt=0;  
    for(int j=0;j<blanks.length;j++)  
    {  
        blanks[j]='-';  
    }  
    secret=secWord.toCharArray();  
  
    for(int i=0;i<blanks.length;i++)  
    {  
        dash+=" - ";  
    }  
  
    lblSecretWord.setText(dash);  
  
    btnStartGame.setVisible(false);  
  
    btnResetGame.setVisible(true);  
  
    for(JButton b : buttons)  
        b.setVisible(true);  
}
```

تصویر (۴۳)

حال متد مربوط به کلیک کردن بر روی حروف و حدس زدن را پیاده سازی میکنیم:

Core Java

آموزش ساده و آسان جاوا

```
private void ClickLetters(java.awt.event.ActionEvent evt) {
    if(!lblStatus.getText().isEmpty()) ۱
    {
        return;
    }

    JButton btn = (JButton)evt.getSource(); ۲
    flag = false;
    help = "";
    win = "";

    String g = "";
    g = btn.getText();
    char guess = g.charAt(0);

    for(int k=0;k<secret.length;k++)
    {
        if(secret[k]==guess)
        {
            flag=true;
            btn.setVisible(false);
            blanks[k]=guess;
        }
    }
}
```

تصویر (۴۴)

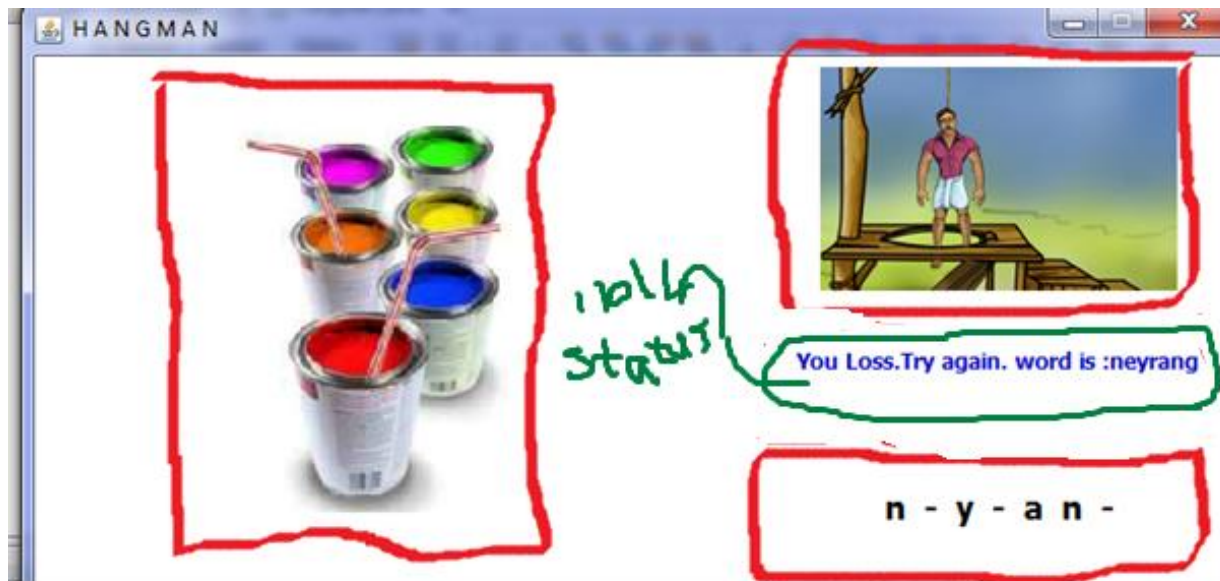
توضیحات قسمت های مشخص شده در تصویر (۴۴):

قسمت ۱:

این شرط باعث میشود که وقتی در lblStatus مقداری ثبت شود (در lblStatus حالت برنده شدن یا بازنده شدن ثبت میشود در تصویر (۴۴) به زیر عکس چوبه دار دقت نمایید بار نگ سبز مشخص شده است)

Core Java

آموزش ساده و آسان جاوا



تصویر (۴۵)

با کلیک کردن بر روی سایر حروف دیگه اتفاقی نیفتد و کار به اتمام رسد.

قسمت ۲:

با این دستور با کلیک کردن بر روی حروف مقدار آن حرف درون دکمه btn ذخیره میگردد.
از آنجا که `evt.getSource()` از نوع `object` میباشد با عمل `cast` کردن به نوع `button` تبدیل میکنیم.

قسمت ۳:

همان متغیرهایی هستند که در مراحل قبل در مورد آنها توضیح داده شد.

Core Java

آموزش ساده و آسان جاوا

قسمت ۴ :

یک متغیر از نوع رشته تعریف کرده ایم. از آنجا که ما با کلیک کردن بر روی حروف مقدار آن حرف مورد نظر ما میباشد با دستور `getText()` مقدارش را درون این متغیر ذخیره میکنیم. از آنجا که حروف الفبا تک کارکتری میباشد با دستور `charAt(0)` به کارکتر تبدیل میکنیم.

قسمت ۵:

هر گاه حدس ما درست باشد با استفاده از حلقه ای که ایجاد کردیم آن حرف درون آرایه `blanks` ذخیره میشود و مقدار `flag` برابر `true` میشود یعنی حدس ما صحیح بوده است و آن دکمه ای که کلیک کردیم با دستور `setVisible(false)` محو میشود. درون حلقه با یک جستجو درون آرایه `secret` که کلمه رمز درون آن است حدس ما را بررسی میکند.

حال ادامه کار مطابق تصویر(۴۵) انجام میدهیم:

Core Java

آموزش ساده و آسان جاوا

```

if(!flag)
{
    btn.setVisible(false);
    lblHangman.setIcon(imgHangman[cnt]);
    cnt++;
}
if(cnt>=7)
{
    lblStatus.setText("You Lose.Try again. word is :"+secWord);
}
for(int z=0;z<blanks.length;z++)
{
    win+=blanks[z];
}
if(win.equals(secWord))
{
    lblStatus.setText("Ok. You win. secret word is :"+secWord);
    lblHangman.setIcon(imgHangman[0]);
}
for(int x=0;x<blanks.length;x++)
{
    help+=blanks[x]+" ";
}
lblSecretWord.setText(help);
}

```

تصویر (۴۶)

توضیحات قسمت های مشخص شده در تصویر (۴۶):

قسمت ۶:

این شرط کنترل میکند که اگر حدس ما درست نبود یعنی مقدار `flag=false` است، واگر نقیض آن درست باشد تصویر مربوط به چوبه دار فعال میشود و یک مقدار به شمارنده `cnt` که تعداد دفعات اشتباه را بررسی میکند اضافه میشود.

قسمت ۷:

اگر تعداد اشتباه به هفت رسید بازنده ایم و پیغام بازنده شدن را نمایش میدهد.

Core Java

آموزش ساده و آسان جاوا

قسمت ۸ :

در هر بار مقدار win که برای تعیین کننده برنده شدن است مقدارش با محتوای آرایه blanks جمع میشود.

قسمت ۹ :

شرطی است که حالت برنده شدن را بررسی میکند.

قسمت ۱۰ :

این بخش برای نمایش حالت خط چین شده کلمه رمز میباشد.

در آخر متد مربوط به Reset_Game را پیاده سازی میکنیم که در این متد همه چیز به حالت اولیه برگردانده میشود.

تصویر(۴۷):

```
private void btnResetGameActionPerformed(java.awt.event.ActionEvent evt) {  
  
    btnResetGame.setVisible(false);  
    btnStartGame.setVisible(true);  
  
    for(JButton b : buttons)  
        b.setVisible(false);  
  
    cnt=0;  
    lblSecretWord.setText("");  
    lblHangman.setIcon(null);  
    lblPicture.setIcon(null);  
    lblStatus.setText("");  
}
```

تصویر(۴۷)

Core Java

آموزش ساده و آسان جاوا

امیدوارم مطالب قابل فهم بوده باشد. با چندبار تکرار و تمرین به خوبی فرا میگیرید.

دوستدار جاوا – سعید قلندری

Core Java

آموزش ساده و آسان جاوا

سایت آموزش زبان جاوا به زبان ساده، آسان و شیرین!!!

www.JAVAPRO.ir

آموزش جاوا SE را با تجربه شفقی و به زبان خودمونی یاد بگیرید!!!!

بازدید از کانال

بازدید از سایت

هر روز مفاهیم و مثال های جدید به سایت اضافه می شود برای اطلاع از مطالب جدید روی سایت عضو کانال شوید.

دفل و تصرف ، ویرایش و کپی زدن تمامی آموزش های جاواالایک به دور از افلاق حرفه ای ست و حرام می باشد.