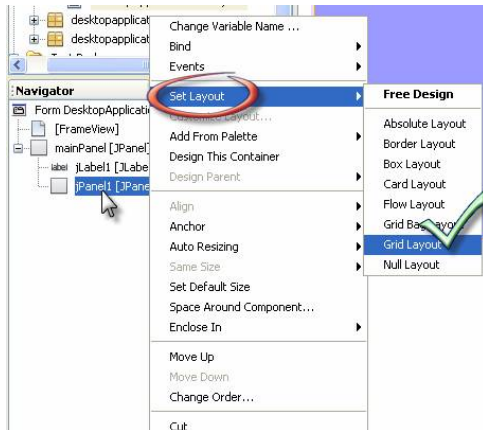
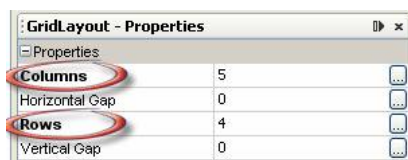


Bài 1. Thiết kế giao diện máy máy tính bỏ túi

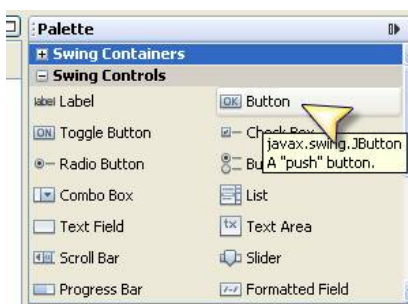
Tạo mới một Project, add một JPanel vào JFrame,

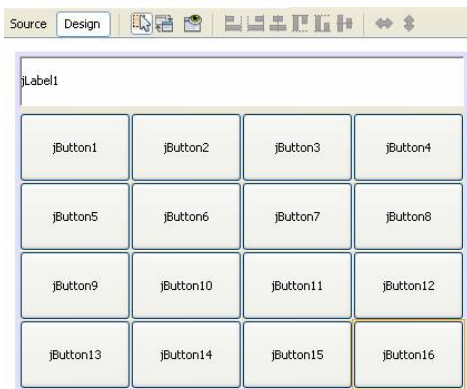


Tiếp theo bạn quy định cho cấu trúc của **GridLayout** là 5 cột, 4 hàng bằng cách Click vào mục **GridLayout** có trong khung **Inspector**, sau đó trong khung **Properties** ở bên phải màn hình của NetBean bạn sẽ quy định thông qua mục **Column** và **Row** như hình dưới đây



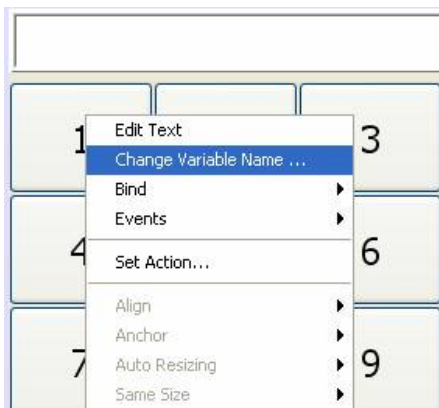
kéo thả vào trong **JPanel** của mình các nút sẽ dùng trên giao diện



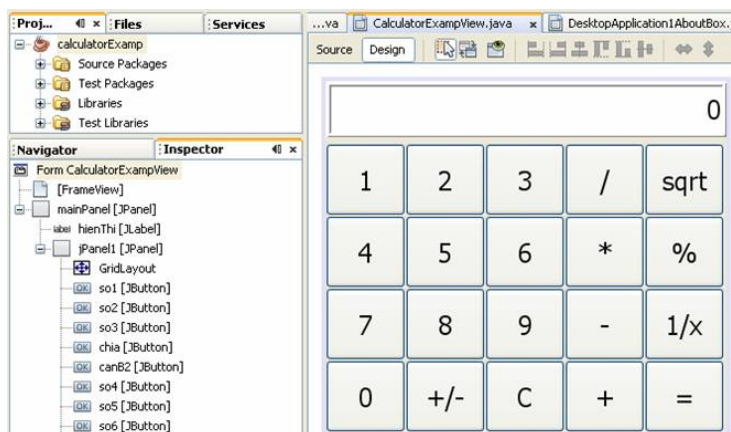


Đặt tên cho các thành phần trên giao diện

Đặt tên cho label dùng để hiển thị dữ liệu là "*hienThi*" bằng cách nhấn nút phải vào **jLabel** này và chọn lệnh "**Change variable Name ...**",



Giao diện hoàn thiện có dạng như sau

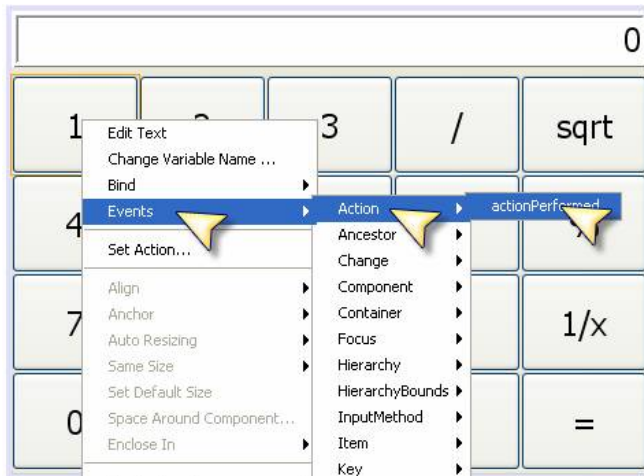


Bài 2. Lập trình sự kiện cho các nút

Lập trình cho các nút số (0,1,2, ... 9)

Để thể hiện số trên JLabel hiển thị trong cửa sổ ứng dụng khi người dùng chọn số nào (nút nào) thì số đó sẽ xuất hiện trong JLabel. Chúng ta sẽ làm như sau

- Trỏ chuột vào nút số cần lập trình và nhấn nút phải, sau đó chọn Events -> Action -> actionPerformed. Các thao tác được mô tả như hình dưới đây



Lúc này, cửa sổ viết Code sẽ xuất hiện. Bạn hãy viết mã lệnh cho sự kiện của nút như sau

```
303 private void solActionPerformed(java.awt.event.ActionEvent evt) {
304     // TODO add your handling code here:
305     String gt=this.hienThi.getText(); //--- Lấy giá trị trong label hiển thị
306     if (gt.length()<12){ //--- Nếu giá trị là số có ít hơn 12 ký số thì ...
307         if (gt.equals("0")) gt=""; //--- Nếu gt là số 0 thì reset thành chuỗi rỗng
308         //--- ghép thêm ký số của nút vào giá trị đã có và đổi ra số nguyên
309         long so = Long.parseLong(gt+ sol.getText());
310         //--- Chuyển số thành chuỗi và gán cho nút vừa nhấn
311         this.hienThi.setText(String.valueOf(so));
312     }
313 }
```

Các nút tương ứng với các phép toán cơ bản (+, -, *, /)

Khi người dùng nhấn vào các nút số để nhập vào số cần tính (**Tại thời điểm này ta xem như họ nhập số thứ nhất**), khi người dùng chọn phép toán chính là lúc bạn phải ghi nhận số đã nhập vào trong 1 biến đồng thời ghi nhận luôn phép toán mà người ta muốn thực hiện, sau đó bạn phải "Reset" lại giá trị trong JLabel hiển thị để cho người dùng có thể tiến hành nhập tiếp giá trị của số thứ 2 có tham gia tính toán trong chương trình. Cho đến khi họ nhấn vào nút "=" thì chúng ta sẽ lấy giá trị thứ nhất (**Đã ghi nhận**) kết hợp với phép toán đã chọn cùng với số thứ 2 (**Số đang có trên JLabel hiển thị tại thời điểm tính toán**) để tính và trả về kết quả trên màn hình. Với phân tích như thế, tôi đã khai báo ở mức toàn cục của lớp View 2 biến thành phần như sau

```
588 //----- Phân khai báo 1 số biến cục bộ dùng cho xử lý
589 private long a = 0; //--- Biến lưu giá trị của số thứ nhất
590 private String phépToan=""; //--- Biến lưu trữ phép toán dùng cho xử lý
591 }
```

mã lệnh của các nút chức năng tương trưng cho 4 phép toán cơ bản như thế này

```

475 private void chiaActionPerformed(java.awt.event.ActionEvent evt) {
476     // TODO add your handling code here:
477     this.a = Long.parseLong(this.hienThi.getText());
478     if (this.a != 0) {
479         this.phepToan = "/";
480         this.hienThi.setText("0");
481     }
482 }
483 private void nhanActionPerformed(java.awt.event.ActionEvent evt) {
484     this.a = Long.parseLong(this.hienThi.getText());
485     this.phepToan = "*"; this.hienThi.setText("0");
486 }
487 private void truActionPerformed(java.awt.event.ActionEvent evt) {
488     this.a = Long.parseLong(this.hienThi.getText());
489     this.phepToan = "-"; this.hienThi.setText("0");
490 }
491 private void congActionPerformed(java.awt.event.ActionEvent evt) {
492     // TODO add your handling code here:
493     this.a = Long.parseLong(this.hienThi.getText());
494     this.phepToan = "+"; this.hienThi.setText("0");
495 }
  
```

Như vậy, sau khi nhấn chọn phép toán và nhập số thứ 2; người dùng chọn nút "=" để xem kết quả của phép toán thì chúng ta sẽ lập trình cho nút này như sau

```

523  /**
524   * Sự kiện thực hiện tính toán giá trị số dựa trên các phép toán cơ bản
525   * @param evt
526   */
527  private void bangActionPerformed(java.awt.event.ActionEvent evt) {
528      // TODO add your handling code here:
529      if (this.phepToan.length() > 0) {
530          //--- Lấy số thứ 2 từ Label hiển thị trên giao diện
531          long b = Long.parseLong(this.hienThi.getText());
532          if (this.phepToan.equals("+")) { //--- Nếu là phép toán cộng thì ...
533              //--- Tính kết quả dựa trên 2 số a và b theo phép toán đã chọn
534              long kq = this.a + b;
535              //--- Chuyển kết quả thành chuỗi và gán cho Label hiển thị
536              this.hienThi.setText(String.valueOf(kq));
537          } else if (this.phepToan.equals("-")) {
538              long kq = this.a - b;
539              this.hienThi.setText(String.valueOf(kq));
540          } else if (this.phepToan.equals("*")) {
541              long kq = this.a * b;
542              this.hienThi.setText(String.valueOf(kq));
543          } else if (this.phepToan.equals("/")) {
544              float kq = (float) this.a / b;
545              this.hienThi.setText(String.valueOf(kq));
546          }
547          //--- Reset phepToan và giá trị của số thứ nhất
548          this.phepToan = ""; this.a = 0;
549      }
550  }

```

www.bodua.com

Tương tự với các phép toán khác