Introduction to Human Computer Interaction

- Livecode Overview

- Based on Livecode User Guide from RunRev Ltd. (2010)

- http://lessons.runrev.com/
Introduction to Human Computer Interaction

- What is LiveCode?
  - A simulation and app development tool for…
    - PCs
    - Macs
    - Mobile Devices
  
- With enough knowledge, you can build most any app

- LiveCode contains
  - A simple set of controls (buttons, fields)
  - An English-like scripting language
  - You do **NOT** need to be a programmer to use LiveCode
Introduction to Human Computer Interaction

- **Event Driven**
  - LiveCode listens for
    - The click off a button
    - The user typing into a field
    - Etc…
  - When an event occurs LiveCode sends a message
    - For example, if a user clicks on a button, LiveCode sends a message to the button.
    - You place code within the button that tells it how to respond to being clicked on.
    - Mouse Events
      - MouseUp
      - MouseDown
      - MouseEnter
Introduction to Human Computer Interaction

- Object Oriented
  - You can start by dragging buttons, text fields, and other controls that make up your application onto a window.
Introduction to Human Computer Interaction

- Object Oriented
  - Once you have the objects in place, you can proceed to attach code to each object to respond to the events you want.
  - Buttons, checkboxes, text fields, menus, graphics, and many more
Introduction to Human Computer Interaction

- **Edit and Run Modes**
  - To enter run mode, choose the browse tool in the *top left of the tools palette*.
  - To edit, choose the pointer tool from the *top right of the tools palette*.
  - Unlike most other development systems, a LiveCode application can be created, edited, debugged and run live.
  - When in run mode, objects receive all the normal messages that drive a LiveCode application.
  - When in edit mode, objects do not receive messages.
Introduction to Human Computer Interaction

- Cards, Stacks & Files
  - The first step in creating a LiveCode application is creating a window, which in LiveCode is called a stack.
  - Each window you see in LiveCode is a stack. Palettes, dialog boxes, and standard windows are all stacks.
  - Each stack contains one or more sets of information called cards.
  - By going from card to card in a stack, you change what's being displayed in that stack's window.
  - You can think of a LiveCode stack as a stack of playing cards, where you can flip through the cards, but only one card at a time is visible.
  - A stack can have a single card or many cards.
Introduction to Human Computer Interaction

- Cards, Stacks & Files
  - Each LiveCode file contains one or more stacks:
    - Either a single main stack, or
    - A main stack and one or more substacks.
    - Since each stack is a window a single stack file can contain multiple windows.
Introduction to Human Computer Interaction

- Cards, Stacks & Files
  - You can create a new Mainstack via the LiveCode menu bar
  - You can open an existing stack via the menu bar
  - The first stack created in a stack file is called the mainstack.
  - Any other stacks created in the same stack file are called substacks of the mainstack.
  - The mainstack contains its substacks
Introduction to Human Computer Interaction

- Other LiveCode Capabilities
  - Media & Resources
    - Importing pictures, movies and video

- External Files and Databases
Introduction to Human Computer Interaction

- **Development Environment**
  - LiveCode is an Integrated Development Environment (IDE).
  - The IDE contains all the features you need to quickly create a professional application.
  - The Application Browser allows you to find your way around your application as you develop it.
  - The Properties Inspector allows you to set appearance and basic behaviors.
  - The Code Editor allows you to add code to each object in your application.
  - In addition to these standard tools, the Message Box provides a mini command-line that allows you to develop aspects of your application automatically or to test out your application’s code and functionality.
Introduction to Human Computer Interaction

- File Menu

![Diagram of Revolution's File Menu]

- New Mainstack
- New Substack
- Open Stack...
- Open Recent File
- Close
- Close and Remove From Memory...
- Import As Control
- New Referenced Control
- Save
- Save As...
- Move Substack to File...
- Revert to Saved...
- Share this Stack...
- Standalone Application Settings...
- Save As Standalone Application...
- Page Setup...
- Print Card...
- Print Field...
- Exit
Introduction to Human Computer Interaction

- **Edit Menu**
Introduction to Human Computer Interaction

- Tools Menu

![Tools Menu](image-url)
Introduction to Human Computer Interaction

- Object Menu
Introduction to Human Computer Interaction

- View Menu
Introduction to Human Computer Interaction

- Help Menu
Introduction to Human Computer Interaction

- **Application Browser**
  - Contains a list of all open stacks, the cards in each stack, and the controls on each card.
  - It allows you to navigate to any card, open or close a stack, select, open the property Inspector for, or edit the script of any object.
  - You can access the Application Browser by choosing **Tools -> Application Browser**
Introduction to Human Computer Interaction

- Application Browser
  - The user view of the application
Introduction to Human Computer Interaction

- Application Browser
  - The user view of the application
Introduction to Human Computer Interaction

- **Application Browser**
  - Right-mouse click functions

![Application Browser](image)
Introduction to Human Computer Interaction

- Application Browser
  - Right-mouse click functions
Introduction to Human Computer Interaction

- The Properties Inspector
  - Different parameters depending on object type

![Properties Inspector](image)
Introduction to Human Computer Interaction

- Code Editor
  - Plain English Scripting

```plaintext
on mouseUp
    put empty into field "Destination Field" of card "Main"
    answer "Order Complete"
end mouseUp
```
Introduction to Human Computer Interaction

- **Code Editor**
  - Learning the language
  - The Code Editor Menu Bar

```plaintext
on mouseUp
    put empty into field "Destination Field" of card "
    answer "Order Complete"
end mouseUp
```
Introduction to Human Computer Interaction

- **Code Editor**
  - What can the language do?

```plaintext
mouseUp

Type: message

Syntax: mouseUp mouseButtonNumber

Objects (or Types): control, card

See Also: click command, clickLoc function, drag message, dragEnd message, mouseDoubleUp message, mouseDown message, mouseRelease message, mouseLoc function, acceleratorModifiers property

Introduced: 1.0
Changed: 1.0

Platform Support: 🌐эмacs

Summary:
Sent when the user releases the mouse button.

Examples:
on mouseUp
  answer "You clicked" & & the name of the target
.end mouseUp
```
Introduction to Human Computer Interaction

- The Message Box
  - Direct communication to your application
  - Command Area
  - Results Area
  - Mode Area
Introduction to Human Computer Interaction

- The Toolbar
  - Quick launch
Introduction to Human Computer Interaction

- Tools Palette
Introduction to Human Computer Interaction

- Alignment and Layers
Introduction to Human Computer Interaction

- The “Ask Question” dialog

Script Editor - button "Suggestion" of card "Main" of stack "D:/Spring 2011 H"

```
.on mouseUp
  ask "Suggestions?"
.end mouseUp
```

Suggestions?

OK Cancel
Introduction to Human Computer Interaction

- The “Answer Alert” dialog

```javascript
on mouseUp
    answer question "What would you like the new flavor to be..." with "Apple" or "Pumpkin" or "Mint" titled "Vote for a Flavor"
end mouseUp
```
Introduction to Human Computer Interaction

- File Selector Dialog

```javascript
on mouseUp
    answer file "Select file:" with type "All Files"
end mouseUp
```

Select file: 

Look in: HCI McKinley
Introduction to Human Computer Interaction

- **Title Bar Selector**

![Title Bar Selector](Image)

- **Stack Details**
  - Name: Ice Cream
  - Title: Ice Cream
  - Main Stack: Ice Cream
  - Controls: default
  - Shape: 0

- **Select a Set of Title Bar Controls**

- **Options**
  - Metal texture
  - Live Resizing
  - Shadow
  - Visible
  - Underline links
  - Format for printing
  - Float above everything
  - Minimized
  - Open minimized
  - Buffer display
  - Purge stack on close
  - Purge window on close
  - Can't delete
  - Can't modify
  - User can't abort scripts

- **Buttons**
  - Cancel
  - OK
Introduction to Human Computer Interaction

- **Button Objects**
  - Push
  - Check boxes & Groups
  - Radio

- **Field Controls**
  - Label
  - Text Entry
  - Scrolling
  - Scrolling List
Introduction to Human Computer Interaction

- Menu Objects
  - Option
  - Pull-Down
  - Combo
  - Pop-Up
Introduction to Human Computer Interaction

- **LiveCode**
  - Writing code is how you give your application functionality.
  - English-like syntax is easy to read and write.
- **A Script**
  - Every object in LiveCode can contain a script, which tells it what to do.
  - Edit the script of an object using the Code Editor
  - Code Editor is a set of individual *message handlers*, each one of which can respond to a different event.
Introduction to Human Computer Interaction

- Handlers
- A complete section of code.
- Four types of handlers, our most useful is the Message Handler
  - Message Handlers
    - Begin with “on” followed by the name of the message that this handler responds to.
    - The handler ends with the “end” and the name of the message.
    - A message handler is executed when the object whose script contains the handler receives the message. This example handler responds to the mouseUp message.
    - Press “Compile” to test script
Events

- LiveCode is based upon events. Every action a script takes is triggered by an event, which is sent in the form of a message.
- When is a Message Sent?
- Messages are sent by events.
- Events include user actions (such as typing a key or clicking the mouse button)
- LiveCode watches for events and sends a message to the appropriate object when an event occurs.
- Examples of built-in messages
  - MouseUp
  - MouseDown
  - MouseStillDown
  - KeyDown
Introduction to Human Computer Interaction

- Responding to Events
  - To respond to an event, you write a message handler with the same name as the message.
  - To respond to a `keyDown` message sent to a field (which is sent when the user presses a key while the insertion point is in the field), place a `keyDown` handler in the field's script:
Introduction to Human Computer Interaction

- Commands and Functions
  - Commands instruct the application to do something – such as play a movie, display a window, or change a property.
Introduction to Human Computer Interaction

- Commands and Functions
  - Functions compute a value
Introduction to Human Computer Interaction

- **Variables**
  - A place to store data that you create, which has no on-screen representation.
  - Can hold any data you want to put into them.
  - Think of a variable as a box with a name on it. You can put anything you want into the box, and take it out later, by simply providing the variable's name.

```plaintext
on menuPick pItemName
  switch pItemName
    Case "Chicken"
      put "Chicken" into field theProtein
    break
    Case "Beef"
      put "Beef" into field theProtein
    break
    Case "Fish"
      put "Fish" into field theProtein
    break
  end switch
end menuPick

on mouseUp
  put "Hello" into introductoryStatement
  put introductoryStatement into field "theEntryField"
end mouseUp

on mouseUp
  global myGlobalVariable
  add 5 to myGlobalVariable
end mouseUp
```
Introduction to Human Computer Interaction

- Containers
  - In addition to variables, LiveCode has six other container types: fields, buttons, images, URLs, the selection, and the message box.
  - Fields, buttons, and imported images are all LiveCode objects. All display their content on the screen.
  - URLs refer to external resources (either files on the system, or items on an Internet server).
  - The Message Box is a special container that's part of the development environment.
  - Literal strings
Introduction to Human Computer Interaction

- Operators
  - Use operators to put together, compare or perform an operation on data.
  - *String Operator* to combines data
  - *Numeric Operator* performs a calculation
  - *Logical Operator* returns true or false
    - Comparison operators (=, <>, <, >, <=, >=)
    - Existence operators (there is a, there is no, is in, is not in, is among, is not among, contains)
    - Basic logical operators (and, or, not)

```plaintext
on mouseUp
  put field "FirstName" into theFirstName
  put ("Hello" && theFirstName & "," && "how have you been?") into msg
end mouseUp

on mouseUp
  put field "theSearchField" into theSearchString
  if theSearchString contains "delaware" then beep
end mouseUp

on mouseUp
  put (1+2) into field "theSolution"
end mouseUp
```
Introduction to Human Computer Interaction

- Making Decisions
  - if...then...else
  - Switch (to choose from a list of options)
Introduction to Human Computer Interaction

- Chunk Expressions
  - An English-like way of describing an exact portion of text. You can use chunks both to retrieve a portion of text, and to edit text.

```plaintext
on mouseUp
   put word 1 of field "myField" into myVariable
end mouseUp

on mouseUp
   put char 1 to 5 of field "myField" into myVariable
end mouseUp

on mouseUp
   put line 1 of field "myField" into myVariable
end mouseUp

on mouseUp
   put word 1 of line 1 of field "myField" into myVariable
end mouseUp
```
Introduction to Human Computer Interaction

- Counting Words and Lines

```plaintext
on mouseUp
    put the number of lines in field "myField" into myVariable
end mouseUp
```
Introduction to Human Computer Interaction

- Referring to Objects
  - In general it is better to refer to an object by its name instead of ID or number

```javascript
on mouseUp
  set the loc of button "OK" to 32,104
end mouseUp
```

- Referring to an objects properties

```javascript
on mouseUp
  set the backgroundColor of button "OK" to blue
end mouseUp
```

- Text properties

```javascript
on mouseUp
  set the textFont of word 3 of field "My Field" to "Courier"
end mouseUp
```
Introduction to Human Computer Interaction

- **Menu Bar**
  - From tools palette
Introduction to Human Computer Interaction

- **Find Command**

  ```on mouseUp
  find "hello world" in field "MyDataField"
  end mouseUp
  ```

- **Printing**

  ```on mouseUp
  answer page setup
  answer printer
  print card
  end mouseUp
  ```
Introduction to Human Computer Interaction

- **Deployment**
  - **Web:** revWeb plugin can deploy web browsers on Windows, Mac OS X and Linux.
    - Save stack as a revlet to run in a browser. You can also deploy desktop revlets that run using the revWeb runtime.
  - **Standalone:** create a native desktop application for each operating system you want to support.
    - Users who do not have LiveCode can run these applications like any other application they download and install.
    - Standalone applications can have their own identity as true applications, include a desktop icon, document associations and more.
Introduction to Human Computer Interaction

- Deployment on Web
Introduction to Human Computer Interaction

- Deployment on Web
Introduction to Human Computer Interaction

- Deployment as Standalone

![Standalone Application Settings for Untitled 1 - Windows](image-url)
Introduction to Human Computer Interaction

- Debugging
Introduction to Human Computer Interaction

- Debugging

![Error message window](image)
Introduction to Human Computer Interaction

- File Management
  - Locating file folders

```
on mouseUp
    global theDocument
    put the filename of this stack into theFileString
    put the length of theFileString into theFileStringLength
    repeat with i = theFileStringLength down to 1
        if char i of theFileString is "/" then
            put i into theSlashPosition
            exit repeat
        end if
    end repeat
    delete char theSlashPosition to theFileStringLength of theFileString
    put theFileString into FolderToPlaceDocuments
    put (theFileString & "/" & theDocument) into theDocumentPathAndName
    launch document theDocumentPathAndName
end mouseUp
```
Introduction to Human Computer Interaction

- Launching other Applications

```javascript
on mouseUp
  launch document "D:\Spring 2011 HCI Course/HCI Schedule Spring 2011.doc"
end mouseUp
```
Introduction to Human Computer Interaction

- Media
  - Importing Images
Introduction to Human Computer Interaction

- Media - Video
  - Playback of video with the player object.
  - On Windows and Mac OS, *QuickTime*, MPEG
Introduction to Human Computer Interaction

- Media - Audio
  - Import audio clip
Introduction to Human Computer Interaction

- Transition Effects

```plaintext
on mouseUp
  visual effect dissolve slow
  go to cd "Kates"
end mouseUp
```
Introduction to Human Computer Interaction

- Paint Tools
Introduction to Human Computer Interaction

- Databases (chapter 8)
  - Database Query Builder
  - RevMedia does not support databases

- RSS Feeds
- iTunes interface
- Other cool stuff…