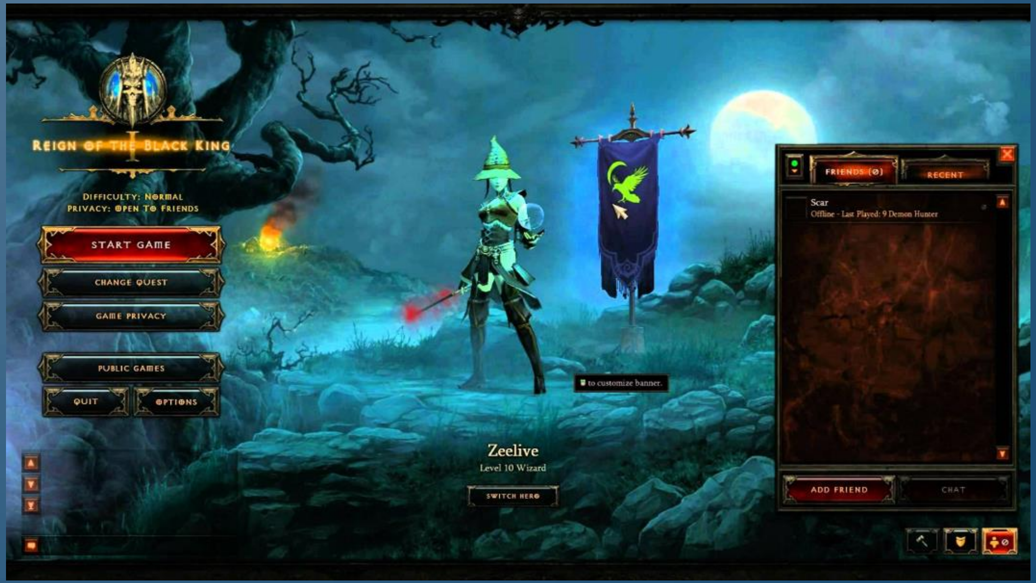
GPAT – CHAPTER 10
USER INTERFACES

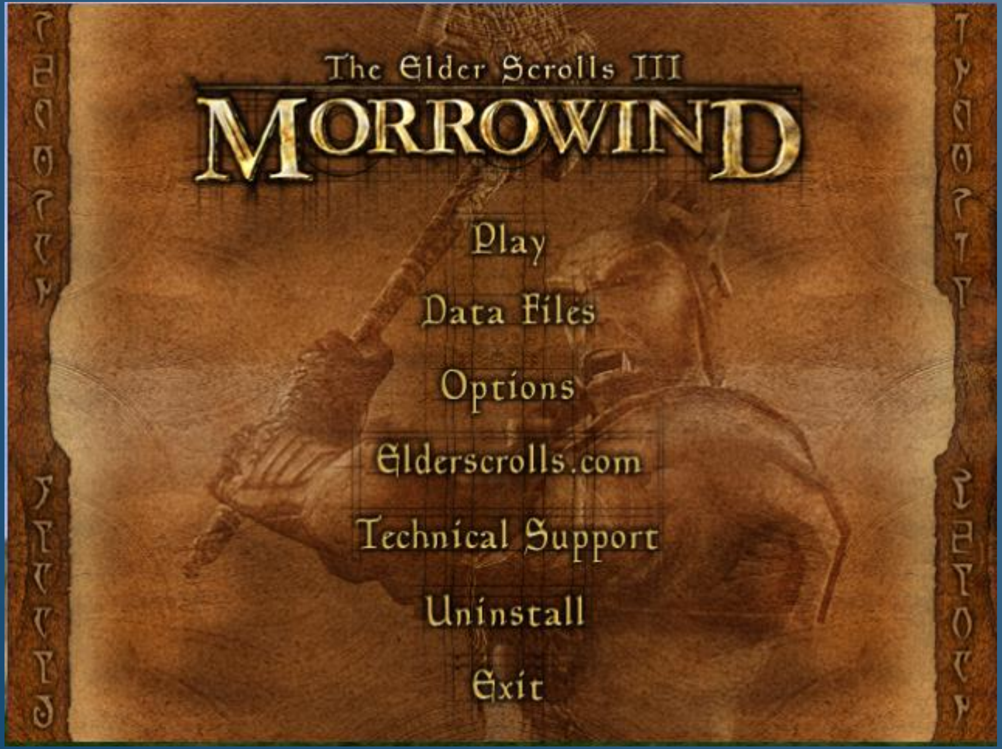
MENU SYSTEMS

MENU STACK

- Ensure a menu system has a stack
  - Need base class for menus
- Allows going back to a prior menu selection
  - Its like allowing a user traversal of a tree
- On entering a new menu – push
- On exiting the current menu – pop
- Good to offer the ability to go to top menu



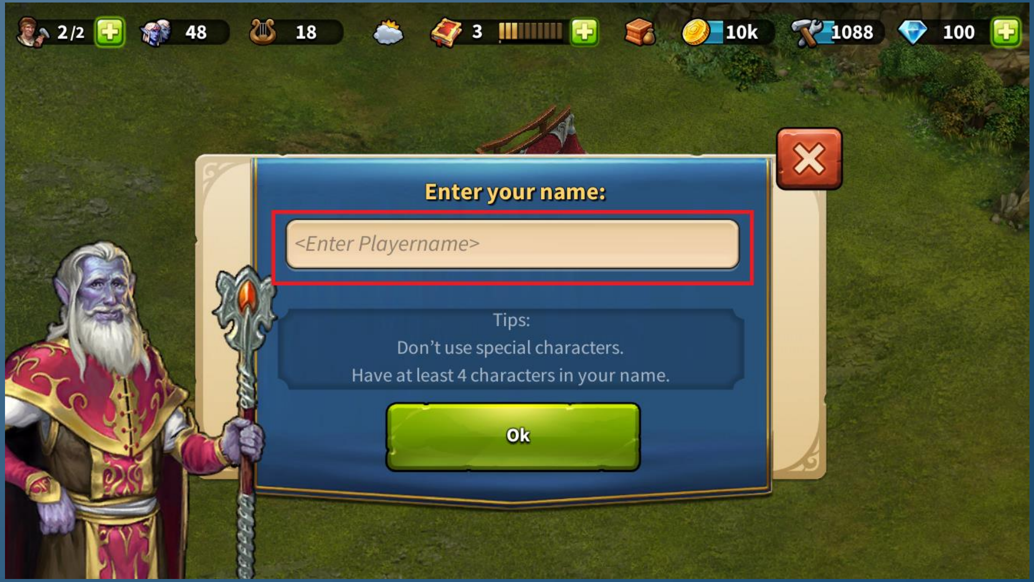
BUTTONS



- Each button needs at least two visual states
  - Selected
  - Unselected
  - Maybe pressed state as well
- Possibly have a doubly-linked list of buttons or an array of buttons tracking an index
  - Back
  - Forward
  - Wrap
- Possibly have a 2D bounding box for clicking or tapping
- Event system to manage what happens when a button is pressed

TYPING

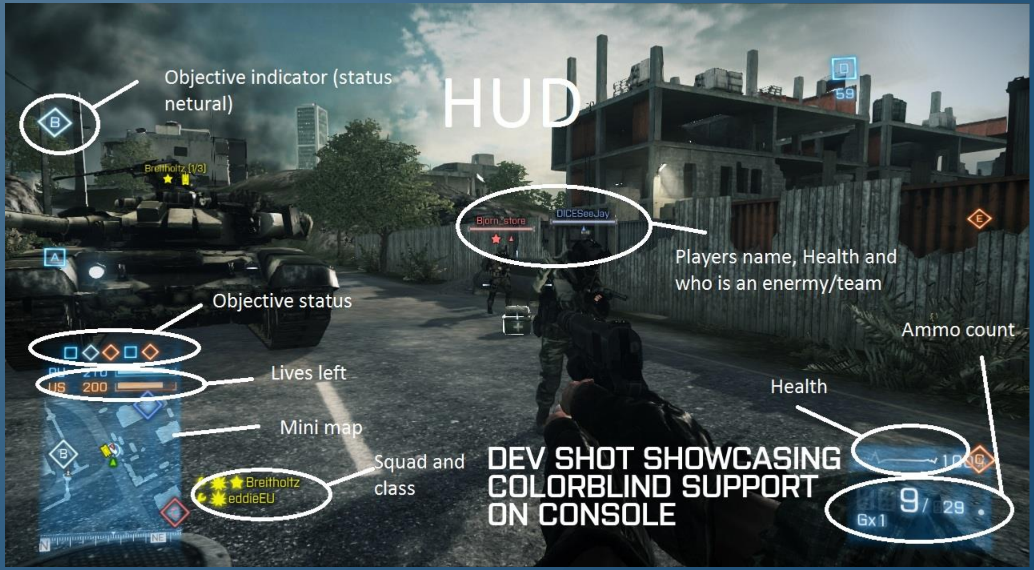
- Supports allowing someone to customize names for avatars
- You can only allow one key stroke at a time, so you construct a string as the player presses keys


2/2
48
18
3
10k
1088
100
Enter your name:
<Enter Playername>
Tips:
Don't use special characters.
Have at least 4 characters in your name.
Ok

HUD ELEMENTS

HUD ELEMENTS

- The HUD or Heads Up Display displays pertinent information to a player for each gameplay mode
- It is an overlay on top of the view of the game
- Simple elements
  - Buttons
  - Score
  - Health bars


HUD
Objective indicator (status netural)
Objective status
Lives left
Mini map
Squad and class
Players name, Health and who is an enermy/team
Ammo count
Health
DEV SHOT SHOWCASING COLORBLIND SUPPORT ON CONSOLE

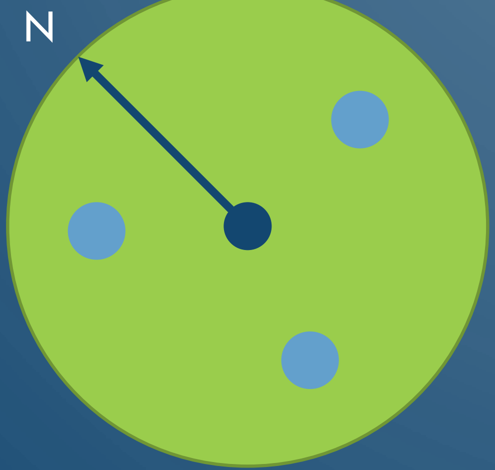
WAYPOINT ARROW

- Store vector for facing direction $\hat{f}$
- Compute vector to target $\hat{t}$
- Angle of rotation
  - $\theta = \cos^{-1}(\hat{t} \cdot \hat{f})$
- Axis of rotation
  - $\vec{a} = \hat{t} \times \hat{f}$
- Rendering considerations
  - Should not be affected by camera transformation
  - Should not be affected by z-buffering

AIMING RETICULE

- Drawn as a crosshair at a set 2D position
- Ray cast is performed into the scene from the unprojected 2D position
- Depending on what the ray hits you change the color/shape of the reticule

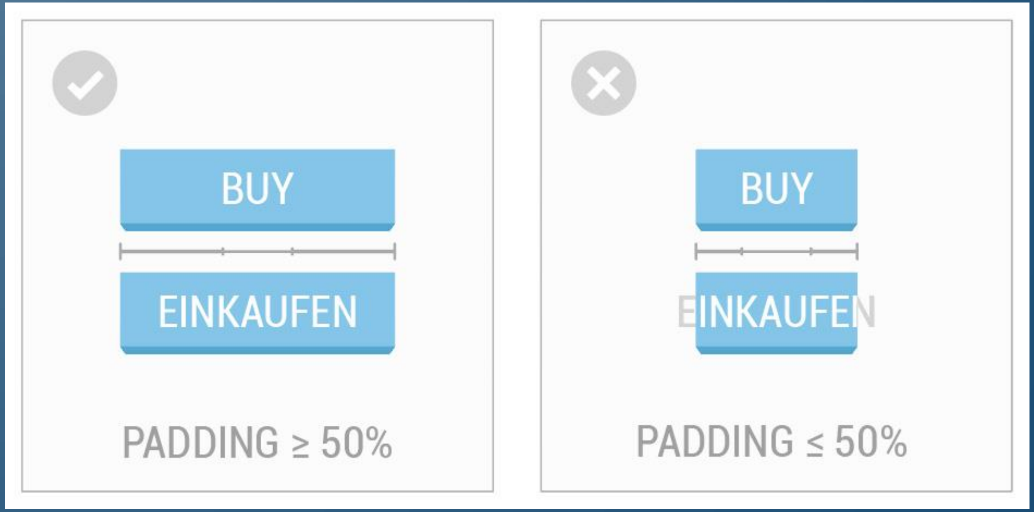
RADAR



- Convert player and objects of interest into 2D positions
- Determine distance and vector to objects of interest
  - Draw blip if inside view based on target vector

OTHER CONSIDERATIONS

- Design in relative coordinates to support multiple resolutions
- Remember to support localization
- Use middleware for the UI as much as possible
- Design for user experience!

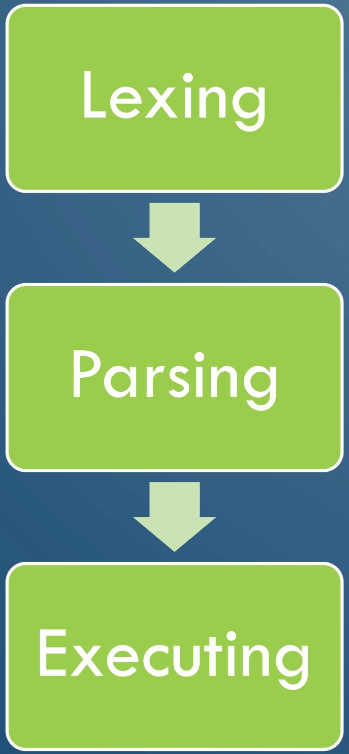


GPAT – CHAPTER 11 (NOT ASSIGNED IN READING) SCRIPTING LANGUAGES AND DATA FORMAT

SCRIPTING LANGUAGES

- Allows designers to get involved in the programming
  - Abstract the engine (hard stuff) from the game elements ("easier" stuff)
- Use a scripting language that is interpreted/compiled by the engine
  - Allows easy updates to the game to be distributed
  - Can reload script dynamically for debugging
  - Prevents crashes
  - However, can be slow

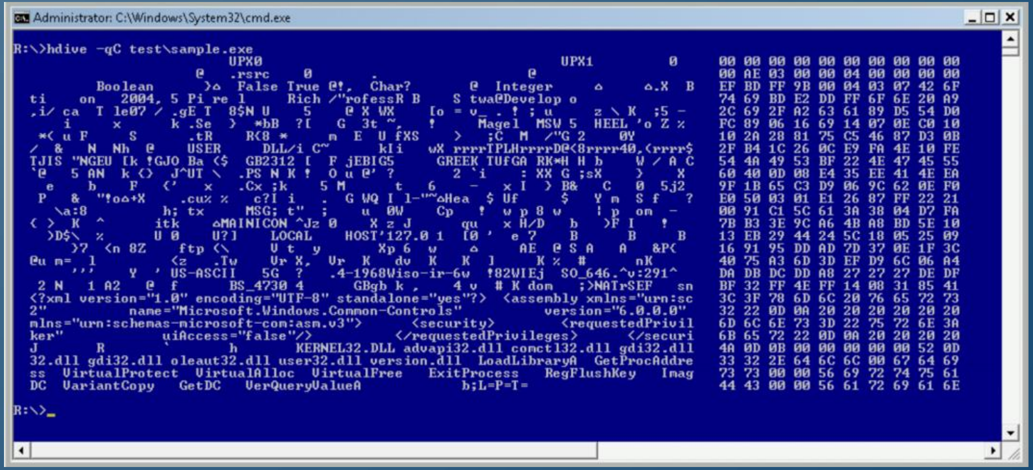
IMPLEMENTING A SCRIPTING LANGUAGE



- Tokenization (Lexical analysis) – make "tokens" out of a stream of text. Typically done through regular expressions.
  - Operators
  - Identifiers
  - Keywords
  - Etc
- Syntax analysis – ensure tokens follow rules of the language. Typically done through context-free grammars.
- Code execution/generation

DATA FORMATS

- Binary file – unreadable file that stores values of the bits directly
  - Efficient
  - Needs some way to help debug/designers
- Text-based file – readable file that stores values of the bits as strings
  - Easy editing for designers and repositories
  - Allows end users to modify (user mods)
  - Can use standard options
    - XML
    - JSON
- Both – text-based in development and binary in release

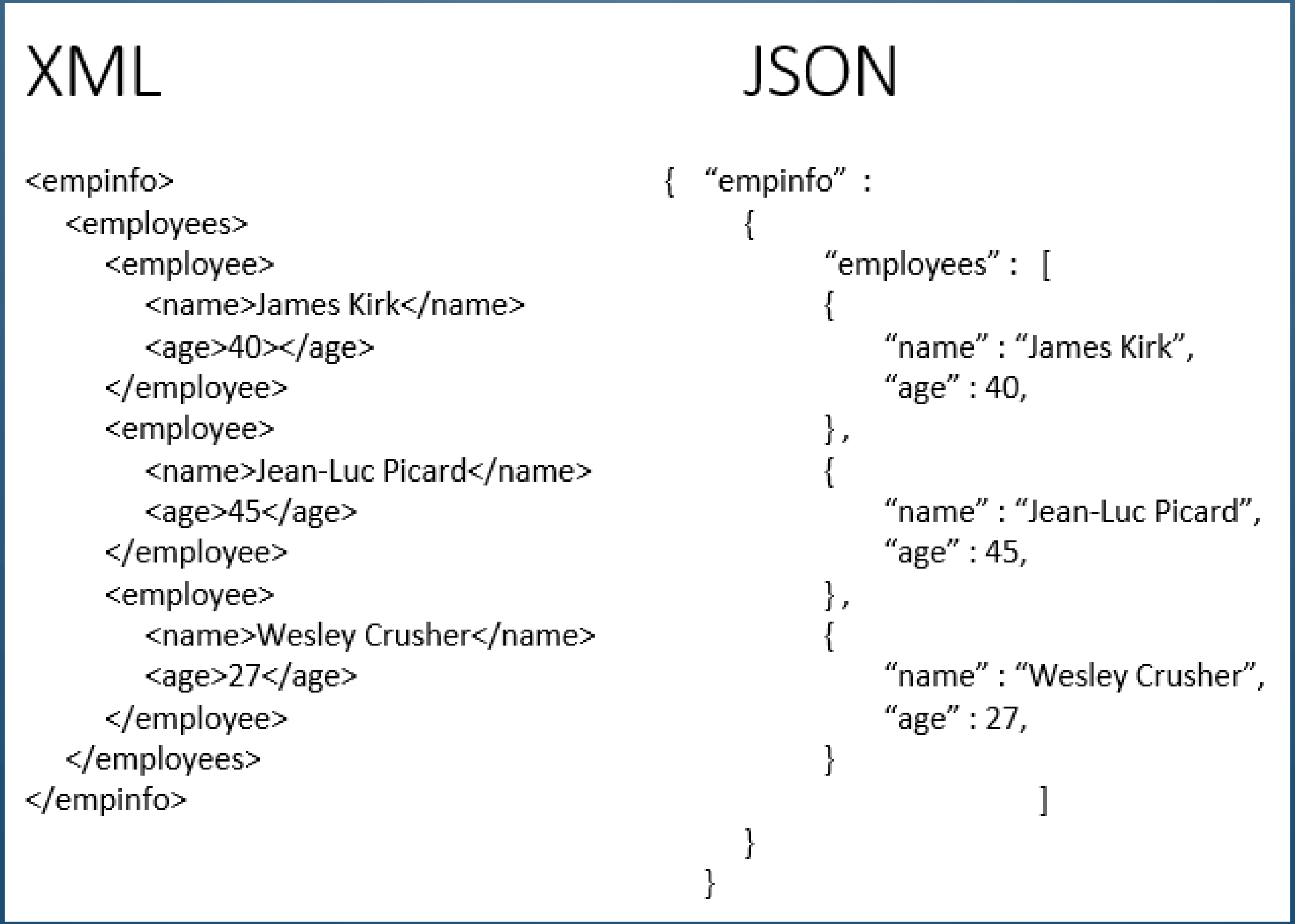


Administrator: C:\Windows\System32\cmd.exe

```
R:\>hdive -qC test\sample.exe
                         UPX0                                 UPX1         0       00 00 00 00 00 00 00 00 00 00
                    @   .rsrc    0       .          @                              00 AE 03 00 00 04 00 00 00 00
        Boolean      }⌂  False True @!,  Char?      @  Integer    ⌂      ⌂.X  B    EF BD FF 9B 00 04 03 07 42 6F
  ti    on   2004, 5 Pi re l     Rich /"rofessR B   S twa@Develop o                74 69 BD E2 DD FF 6F 6E 20 A9
  ,i/ ca  T le07 / .gE T  8$N U    5    @ X WX   [o = v_ . ! ; u     z \ K  ;5 -   2C 69 2F A2 63 61 89 D5 54 D0
     i    x      k .Se  }  *bB  ?[   G  3t ~,   !    Magel  MSW 5  HEEL 'o Z %    FC 89 06 16 69 14 07 0E C0 10
   *( u F    S      .tR    R<8 *     m  E  U fXS    >   ;C  M  /"G 2     0Y       10 2A 28 81 75 C5 46 87 D3 0B
  /  &   N  Nh  @    USER     DLL/i C~      kIi   wX rrrrTPLHrrrrD@<8rrrr40,<rrrr$ 2F B4 1C 26 0C E9 FA 4E 10 FE
  TJIS "NGEU [k !GJO Ba <$  GB2312 [  F jEBIG5     GREEK TUfGA RK*H H b    W / A C 54 4A 49 53 BF 22 4E 47 45 55
  `@   5 AN  k <>  J^UT \  .PS N K !  O u @' ?      2 `i    : XX G ;sX    }     X  60 40 0D 08 E4 35 EE 41 4E EA
   e    b    F   <'   x   .Cx ;k    5 M       t   6     -   x I  } B&   C   0  5j2 9F 1B 65 C3 D9 06 9C 62 0E F0
   P   &  "!o⌂+X     .cu% %   c?I i   .  G WQ I 1-'"~⌂Hea  $ Uf      $   Y m  S f   ? E0 50 03 01 E1 26 87 FF 22 21
    \a:8          h; tx     MSG; t"  ;     u  0W     Cp   !  w p 8 w    ¦ p  om  - 00 91 C1 5C 61 3A 38 04 D7 FA
  < >  K  ^        itk    ⌂MAINICON ^Jz 0    X z J       qu   x H/D    b   }F I   ! 7B B3 3E 9C A6 4B A8 BD 5E 10
   }D$\  %       U 0    U?]    LOCAL    HOST'127.0 1  [0 '  e 7    B       B     B 13 EB 29 44 24 5C 18 05 25 09
      }7  <n 8Z     ftp <\       U t  y          Xp 6  w     ⌂     AE  @ S A   A  &P< 16 91 95 DD AD 7D 37 0E 1F 3C
  @u m=  l          <z   .Tw     Ur X,  Ur  K   dv  K    K  ]     K %  #      nK   40 75 A3 6D 3D EF D9 6C 06 A4
      '''     Y  ' US-ASCII   5G  ?   .4-1968Viso-ir-6w  !82WIEj  SO_646.^v:291^   DA DB DC DD A8 27 27 27 DE DF
   2 N   1 A2   @  f       BS_4730 4      GBgb k ,    4 v  # K dom   ;>NATrSEF  sn BF 32 FF 4E FF 14 08 31 85 41
  <?xml version="1.0" encoding="UTF-8" standalone="yes"?>  <assembly xmlns="urn:sc 3C 3F 78 6D 6C 20 76 65 72 73
  2"         name="Microsoft.Windows.Common-Controls"            version="6.0.0.0" 32 22 0D 0A 20 20 20 20 20 20
  mlns="urn:schemas-microsoft-com:asm.v3">       <security>        <requestedPrivil 6D 6C 6E 73 3D 22 75 72 6E 3A
  ker"            uiAccess="false"/>           </requestedPrivileges>     </securi 6B 65 72 22 0D 0A 20 20 20 20
  J        R          `      h          KERNEL32.DLL advapi32.dll comctl32.dll gdi32.dll 4A 0D 0B 00 00 00 00 00 52 0D
  32.dll gdi32.dll oleaut32.dll user32.dll version.dll  LoadLibraryA  GetProcAddre 33 32 2E 64 6C 6C 00 67 64 69
  ss  VirtualProtect  VirtualAlloc  VirtualFree   ExitProcess   RegFlushKey   Imag 73 73 00 00 56 69 72 74 75 61
  DC  VariantCopy   GetDC   VerQueryValueA             b;L=P=T=                    44 43 00 00 56 61 72 69 61 6E

R:\>_
```

DATA FORMATS



XML

```
<empinfo>
   <employees>
      <employee>
         <name>James Kirk</name>
         <age>40></age>
      </employee>
      <employee>
         <name>Jean-Luc Picard</name>
         <age>45</age>
      </employee>
      <employee>
         <name>Wesley Crusher</name>
         <age>27</age>
      </employee>
   </employees>
</empinfo>
```

JSON

```
{   “empinfo” :
        {
              “employees” :  [
              {
                   “name” : “James Kirk”,
                   “age” : 40,
              },
              {
                   “name” : “Jean-Luc Picard”,
                   “age” : 45,
              },
              {
                   “name” : “Wesley Crusher”,
                   “age” : 27,
              }
                                 ]
        }
   }
```

WORK ON DESIGNING YOUR MENU SYSTEM AND HUD FOR YOUR PRIMARY GAMEPLAY MODES

SUMMARY

- In this chapter, we looked at some basic approaches to defining and implementing a user interface for a game
  - Menu systems
  - HUD elements