CS 50 Walkthrough 7
Problem Set 7: C$50 Finance

Marta Bralic
(slides courtesy of Keito)
Problem Set 7: C$50 Finance

• Topics:
  – Web programming
  – PHP
  – MySQL

• You implement:
  – A stock buying/selling/portfolio viewing/history viewing website
Permissions

Read: \( r = 100 \) (4)
Write: \( w = 010 \) (2)
Execute: \( x = 001 \) (1)

rwx \( (4 + 2 + 1 = 7) \)
rw_ \( (4 + 2 + 0 = 6) \)
r_x \( (4 + 0 + 1 = 5) \)
r__ \( (4 + 0 + 0 = 4) \)
_wx \( (0 + 2 + 1 = 3) \)
_w_ \( (0 + 2 + 0 = 2) \)
__x \( (0 + 0 + 1 = 1) \)

Owner/Group/Others
rwx rwx rwx \( = 777 \)
r_x r_x x__ \( = 551 \)
rw_ ___ rw_ \( = 606 \)
Permissions on the web

• **Static content** (.html, .png, .jpg, .css, .js, etc.): readable by “others”, read-write by “owner”
  – chmod 644 name

• **Directories** (including your home directory, public_html, and any directories within): executable by “others”, read-write-execute by “owner”
  – chmod 711 name

• **PHP scripts** (.php): read-write by “owner”
  – chmod 600 name
Distribution code

- index.php – homepage
- common.php – included in index.php
- stock.php – class for stock, similar to C struct
- login.php – form for logging in via POST
- login2.php – starts session
- helpers.php – lookup(), dump(), apologize()
Features you need to implement

• User registration
• Get quotes
• Maintaining a Portfolio
• Buying
• Selling
• History
SQL

• Login to database

• Three “steps” to manipulating tables
  – $query = create query with INSERT, SELECT, UPDATE or DELETE
  – $result = mysql_query($query)
  – $row = mysql_fetch_array($result)
    • $row[“field”]
register.php

• copy login.php and login2.php using cp
  – change head’s title
  – change login2.php to register2.php
  – add password2 field
  – change submit button to say register
  – change link to login
register2.php (pseudocode)

• If any field is blank or password and password2 don’t match
  – apologize

• Otherwise
  – $sql = INSERT INTO users table...
  – $result = mysql_query($sql);
  – if NULL
    • apologize!
  – else
    • log them in
Get Quotes!

• quote.php
  – <form action="quote2.php" method="post">
  – similar to login.php

• quote2.php
  – lookup
  – print
Additional Databases

• Portfolio:
  – uid (user ID of owner)
  – symbol (of stock)
  – shares (quantity)

• History
  – uid
  – bought or sold
  – shares and price
  – date and time
Buy!

- buy.php
  - `<form action="buy2.php" method="post">`
    - need stock symbol, and number of shares

- buy2.php
  - `$uid = $_SESSION["uid"];` stores current user
  - add stock to portfolio
    - INSERT like in register2.php but into portfolio table
  - charge user
    - UPDATE user table
  - log transaction in history
    - INSERT into history table
Buy Continued

• look up price of stock (like in quote)
  – lookup($symboltobuy); is stock “struct”
    • need to get the cost from this struct

• look up user’s cash
  – if they can’t afford it
    • apologize!
  – else
    • complete the transaction
Sell!

- Extremely similar to buy
- If user owns the stock
  - DELETE FROM portfolio...
  - refund their money
    - look up price * number of shares
  - log in history table
History

- $history = SELECT FROM history....
- while ($row=mysql_fetch_array($result))
  print values out
When you think you’re done…

- Does your site work across browsers?
- Is your input validation sufficient (very important!)
  - Empty values
  - Negative shares
  - Fractional shares
  - etc.
- Do you check for failure (NULL) of, say, `mysql_query()`?
CS 50 Walkthrough 7
Problem Set 7: C$50 Finance

Marta Bralic