

ANSWER KEY

1. PHP

```
<!-- order.php -->
<form action="order-submit.php" method="post">
  <div>
    Food item:
    <select name="food">
      <?php foreach (glob("*.jpg") as $image) { ?>
        <option> <?= str_replace(".jpg", "", $image) ?> </option>
      <?php } ?>
    </select>
  </div>
  <div> Quantity: <input type="text" name="quantity" size="2" /> </div>
  <div> <input type="submit" value="Order" /> </div>
</form>

<?php
# order-submit.php
if (!isset($_POST["food"]) || !isset($_POST["quantity"])) {
  header("HTTP/1.1 400 Invalid Request");
  die();
}

$food = $_POST["food"];
$quantity = $_POST["quantity"];
$found = FALSE;
foreach (file("inventory.txt") as $line) {
  list($item, $number, $price) = explode("\t", $line);      # list() is optional
  if ($item == $food && $number >= $quantity) {
    $total = $quantity * $price;
    ?>

  <p>
    Order successful!
    $<?= $total ?> is your total price.
    Here is what you ordered:

    <?php for ($i = 0; $i < $quantity; $i++) { ?>
      
    <?php } ?>
  </p>

  <?php
  $found = TRUE;
  break;                                                    # optional
}
}

if (!$found) { ?>
  <p>Sorry, we don't have <?= $quantity ?> of '<?= $food ?>' in stock.</p>
<?php } ?>
```

2. JavaScript / DOM

```
window.onload = function() {
    document.getElementById("find").onclick = findClick;
};

function findClick() {
    document.getElementById("palindromes").innerHTML = "";
    var words = document.getElementById("phrase").value.split(" ");
    var count = 0;
    for (var i = 0; i < words.length; i++) {
        if (document.getElementById("min").value &&
            words[i].length < document.getElementById("min").value) { continue; }
        if (document.getElementById("max").value &&
            words[i].length > document.getElementById("max").value) { continue; }
        if (isPalindrome(words[i])) {
            var li = document.createElement("li");
            li.innerHTML = words[i];
            if (count % 2 == 0) {
                li.style.backgroundColor = "#cccccc";
            }
            document.getElementById("palindromes").appendChild(li);
            count++;
        }
    }
    document.getElementById("count").innerHTML = count + " total palindrome(s).";
}

function isPalindrome(s) {
    s = s.toLowerCase();
    for (var i = 0; i < s.length / 2; i++) {
        if (s[i] != s[s.length - 1 - i]) { return false; }
    }
    return true;
}
```

3. Regular Expressions

- a) `/^[a-zA-Z]{4}$/`
- b) `/^[3[47]][0-9]{13}$/`
- c) `/^.*\./.*\.(html|css|js|php)$/`

4. SQL

```
-- NOTE: *DON'T* run this query on imdb; it takes too long and hangs.  
-- Run it on imdb_small only, please.  
-- all directors who have made >= 2 movies across >= 2 genres
```

```
SELECT DISTINCT d.id, d.last_name d.first_name  
FROM directors d  
JOIN movies_directors md1 ON md1.director_id = d.id  
JOIN movies_genres mg1 ON mg1.movie_id = md1.movie_id  
JOIN movies_directors md2 ON md2.director_id = d.id  
JOIN movies_genres mg2 ON mg2.movie_id = md2.movie_id  
WHERE mg1.genre <> mg2.genre AND mg1.movie_id <> mg2.movie_id  
ORDER BY d.last_name, d.first_name;
```

```
-- all directors who have made >= 2 movies across >= 2 genres  
-- (includes movies table unnecessarily)
```

```
SELECT DISTINCT dir.id, dir.last_name, dir.first_name  
FROM directors dir  
JOIN movies_directors md1 ON md1.director_id = dir.id  
JOIN movies m1 ON m1.id = md1.movie_id  
JOIN movies_genres mg1 ON mg1.movie_id = m1.id  
JOIN movies_directors md2 ON md2.director_id = dir.id  
JOIN movies m2 ON m2.id = md2.movie_id  
JOIN movies_genres mg2 ON mg2.movie_id = m2.id  
WHERE m1.id <> m2.id AND mg1.genre <> mg2.genre  
ORDER BY dir.last_name, dir.first_name;
```