```
<!-- a webpage on fact families, including integers, fractions,
decimals, and algebraic equations -->

<script src='externalJS/random_number_generator.js'>
</script>

<script src='externalJS/round_numbers.js'>
</script>

<script src='externalJS/gcf lcm.js'>
</script>

<script>
// global variables
var term1, term2, sum, factor1, factor2, product;
var type; // wholeNumber, integer, fraction, decimal, oneStep, linear,
literal
var fact1, fact2, fact3, fact4, givenFact; // the four facts of the
family, and the fact given to user
var temp1; // temporary variable
var operation; // either addSub or multDiv
var problem; // the text displayed for each problem

var fractions = [ // '12' means 1/2, etc.
    '12', '13', '23', '14', '34', '15', '25', '35', '45', '16', '56',
'18', '38', '58', '78'
]

// main function
function getNewProblem() {

    resetPage();
    getProblemType(); // either add/sub or mult/div; then select
between type of fact family
    displayGivenFact();
    getFacts();


} // end of getNewProblem()

// reset the page display when a new problem is generated
function resetPage() {
    document.getElementById('errorMessageDiv').style.visibility =
'hidden';
    document.getElementById('answerSpace').innerHTML = '';
} // end of resetPage()

// determine which radio button is selected and get the appropriate
problem type
function getProblemType() {
    // addition/subtraction or multiplication/division?
    temp1 = getNumber(0, 1);
    if (temp1===0) {operation = 'addSub';}
    else {operation = 'multDiv';}

    if (document.getElementById('wholeNumbersRadio').checked)
```

```
{type='wholeNumber'; getWholeNumber();}
    else if (document.getElementById('integersRadio').checked)
{type='integer'; getInteger();}
    else if (document.getElementById('fractionsRadio').checked)
{type='fraction'; getFraction();}
    else if (document.getElementById('decimalsRadio').checked)
{type='decimal'; getDecimal();}
    else if (document.getElementById('oneStepRadio').checked)
{type='oneStep'; getOneStep();}
    else if (document.getElementById('linearRadio').checked)
{type='linear'; getLinear();}
    else if (document.getElementById('literalRadio').checked)
{type='literal'; getLiteral();}
    else {document.getElementById('errorMessageDiv').style.visibility
= 'visible';}
} // end of getProblemType()

// get a fact family using whole numbers
function getWholeNumber() {
    if (operation==='addSub') {
        term1 = getNumber(0, 20);
        term2 = getNumber(0, 19); if (term2===term1) {term2 = 20;}
        sum = term1 + term2;

        // randomly select which fact to give
        temp1 = getNumber(0, 3);
        switch (temp1) {
            case 0: givenFact = term1 + ' + ' + term2 + ' = ?'; break;
            case 1: givenFact = term2 + ' + ' + term1 + ' = ?'; break;
            case 2: givenFact = sum + ' &ndash; ' + term1 + ' = ?';
break;
            case 3: givenFact = sum + ' &ndash; ' + term2 + ' = ?';
break;
            default: alert('error - getWholeNumber()');
        }
    }
    else {
        factor1 = getNumber(1, 12);
        factor2 = getNumber(1, 11); if (factor2===factor1) {factor2 =
12;}
        product = factor1*factor2;

        // randomly select which fact to give
        temp1 = getNumber(0, 3);
        switch (temp1) {
            case 0: givenFact = '(' + factor1 + ')(' + factor2 + ') =
?'; break;
            case 1: givenFact = '(' + factor2 + ')(' + factor1 + ') =
?'; break;
            case 2: givenFact =  product + ' / ' + factor1 + ' = ?';
break;
            case 3: givenFact =  product + ' / ' + factor2 + ' = ?';
break;
            default: alert('error - getWholeNumber()');
        }
    }
} // end of getWholeNumber()
```

```
// fact family with integers
function getInteger() {
    if (operation==='addSub') {
        term1 = getNumber(-20, 20);
        term2 = getNumber(-20, 19); if (term1===term2) {term2 = 20;}
        sum = term1 + term2;

        temp1 = getNumber(0, 3);
        switch(temp1) {
            case 0: givenFact = '(' + term1 + ') + (' + term2 + ') =
?'; break;
            case 1: givenFact = '(' + term2 + ') + (' + term1 + ') =
?'; break;
            case 2: givenFact = '(' + sum + ') &ndash; (' + term1 + ')
= ?'; break;
            case 3: givenFact = '(' + sum + ') &ndash; (' + term2 + ')
= ?'; break;
            default: alert('error - getInteger()');
        }
    }
    else {
        factor1 = getNumber(-12, 12);
        factor2 = getNumber(-12, 11); if (factor1===factor2) {factor2
= 12;}
        product = factor1*factor2;

        temp1 = getNumber(0, 3);
        switch (temp1) {
            case 0: givenFact = '(' + factor1 + ')(' + factor2 + ') =
?'; break;
            case 1: givenFact = '(' + factor2 + ')(' + factor1 + ') =
?'; break;
            case 2: givenFact = '(' + product + ') / (' + factor1 + ')
= ?'; break;
            case 3: givenFact =  '(' + product + ') / (' + factor2 +
') = ?'; break;
            default: alert('error - getInteger()');
        }
    }
} // end of getInteger()

// fact family with fractions
function getFraction() {
    var num1, num2, den1, den2, sumNum, sumDen, productNum,
productDen;
    temp1 = getNumber(0, fractions.length-1);
    num1 = fractions[temp1].slice(0,1);
    den1 = fractions[temp1].slice(1,2);
    temp2 = getNumber(0, fractions.length-2); if (temp1===temp2)
{temp2=fractions.length-1;}
    num2 = fractions[temp2].slice(0,1);
    den2 = fractions[temp2].slice(1,2);

    if (operation==='addSub') {
        term1 = '<div class="fraction"><span class="fup">' + num1 +
'</span><span class="fdn">';
```

```
        term1 += den1 + '</span></div>';
        term2 = '<div class="fraction"><span class="fup">' + num2 +
'</span><span class="fdn">';
        term2 += den2 + '</span></div>';
        sumNum = num1*den2 + num2*den1;
        sumDen = den1*den2;
        temp1 = hcf(sumNum, sumDen); // external js; get the greatest
common denominator
        sumNum /= temp1;
        sumDen /= temp1;
        if (sumNum % sumDen == 0) {sum = sumNum / sumDen;}
        else {
            sum = '<div class="fraction"><span class="fup">' + sumNum
+ '</span>';
            sum += '<span class="fdn">' + sumDen + '</span></div>';
        }

        temp1 = getNumber(0, 3);
        switch(temp1) {
            case 0: givenFact = term1 + ' + ' + term2 + ' = ?'; break;
            case 1: givenFact = term2 + ' + ' + term1 + ' = ?'; break;
            case 2: givenFact = sum + ' &ndash; ' + term1 + ' = ?';
break;
            case 3: givenFact = sum + ' &ndash; ' + term2 + ' = ?';
break;
            default: alert('error - getInteger()');
        }
    }
    else { // multiply and divide
        factor1 = '<div class="fraction"><span class="fup">' + num1 +
'</span>';
        factor1 += '<span class="fdn">' + den1 + '</span></div>';
        factor2 = '<div class="fraction"><span class="fup">' + num2 +
'</span>';
        factor2 += '<span class="fdn">' + den2 + '</span></div>';
        productNum = num1*num2;
        productDen = den1*den2;
        temp1 = hcf(productNum, productDen);
        productNum /= temp1;
        productDen /= temp1;
        product = '<div class="fraction"><span class="fup">' +
productNum + '</span>';
        product += '<span class="fdn">' + productDen + '</span>
</div>';

        temp1 = getNumber(0, 3);
        switch (temp1) {
            case 0: givenFact = factor1 + ' &sdot; ' + factor2 + ' =
?'; break;
            case 1: givenFact = factor2 + ' &sdot; ' + factor1 + ' =
?'; break;
            case 2: givenFact = product + ' &divide; ' + factor1 + ' =
?'; break;
            case 3: givenFact =  product + ' &divide; ' + factor2 + '
= ?'; break;
            default: alert('error - getInteger()');
        }
```

```
    }
} // end of getFraction()

// fact family with decimals
function getDecimal() {
    // each add/sub problem has two terms with decimals in different
positions from right
    operation = 'addSub';
    if (operation==='addSub') {
        term1 = getNumber(1, 99);
        term2 = getNumber(1, 98); if (term1===term2) {term2 = 99;}
        switch (term2) {
            case 10: term2 = 89; break;
            case 20: term2 = 91; break;
            case 30: term2 = 92; break;
            case 40: term2 = 93; break;
            case 50: term2 = 94; break;
            case 60: term2 = 95; break;
            case 70: term2 = 96; break;
            case 80: term2 = 97; break;
            case 90: term2 = 98; break;
        }
        if (term2===term1) {term2 = 99;}

        // randomly select how many digits the first term has to the
right of the decimal
        temp1 = getNumber(0, 2);
        temp2 = Math.pow(10, temp1);
        term1 /= temp2;
        term1 = roundDecimalNumber(term1, temp1);
        switch (temp1) {
            case 0:
            case 1: // second term can either be to hundredths or
tenths
                temp2 = getNumber(0, 1);
                if (temp2===0) {
                    term2 /= 10;
                    term2 = roundDecimalNumber(term2, 1);
                }
                else {
                    term2 /= 100;
                    term2 = roundDecimalNumber(term2, 2);
                }
                break;
            case 2:
                term2 /= 100;
                term2 = roundDecimalNumber(term2, 2);
                break;
            default: alert('error - getDecimal()');
        }

        sum = term1 + term2;

        temp1 = getNumber(0, 3);
        switch(temp1) {
            case 0: givenFact = term1 + ' + ' + term2 + ' = ?'; break;
            case 1: givenFact = term2 + ' + ' + term1 + ' = ?'; break;
```

```
            case 2: givenFact = sum + ' &ndash; ' + term1 + ' = ?';
break;
            case 3: givenFact = sum + ' &ndash; ' + term2 + ' = ?';
break;
            default: alert('error - getInteger()');
        }
    }
    // each mult/div problem is two digits times two digits (at most)
} // end of getDecimal()

// fact family as one-step equation with one variable
function getOneStep() {

} // end of getOneStep()

// fact family as linear equation with two variables
function getLinear() {

} // end of getLinear()

// fact family of literal equations
function getLiteral() {

} // end of getLiteral()

// display one fact for the user
function displayGivenFact() {
    // slightly different instructions for facts with only #'s vs.
"facts" involving variables
    if (type==='wholeNumber' || type==='integer' || type==='fraction'
|| type==='decimal') {
        problem = 'Complete the given math fact.<br />Then give all
four facts in this fact family.';
    }
    else {problem = 'One fact of a fact family is given.<br />Give all
four facts in this fact family.';}

    problem += '<br /><br />' + givenFact;

    document.getElementById('problemSpace').innerHTML = problem;
} // end of displayGivenFact()

// get the four facts for the family based on terms and sum or factors
and product
function getFacts() {
    if (type==='wholeNumber') {
        if (operation==='addSub') {
            fact1 = term1 + ' + ' + term2 + ' = ' + sum;
            fact2 = term2 + ' + ' + term1 + ' = ' + sum;
            fact3 = sum + ' &ndash; ' + term1 + ' = ' + term2;
            fact4 = sum + ' &ndash; ' + term2 + ' = ' + term1;
        }
        else {
            fact1 = '(' + factor1 + ')(' + factor2 + ') = ' + product;
            fact2 = '(' + factor2 + ')(' + factor1 + ') = ' + product;
            fact3 = '<div class="fraction"><span class="fup">' +
product;
```

```
            fact3 += '</span><span class="fdn">' + factor1 + '</span>
</div> = ' + factor2;
            fact4 = '<div class="fraction"><span class="fup">' +
product;
            fact4 += '</span><span class="fdn">' + factor2 + '</span>
</div> = ' + factor1;
        }
    }
    else if (type==='integer') {
        if (operation==='addSub') {
            fact1 = '(' + term1 + ') + (' + term2 + ') = ' + sum;
            fact2 = '(' + term2 + ') + (' + term1 + ') = ' + sum;
            fact3 = '(' + sum + ') &ndash; (' + term1 + ') = ' +
term2;
            fact4 = '(' + sum + ') &ndash; (' + term2 + ') = ' +
term1;
        }
        else {
            fact1 = '(' + factor1 + ')(' + factor2 + ') = ' + product;
            fact2 = '(' + factor2 + ')(' + factor1 + ') = ' + product;
            fact3 = '<div class="fraction"><span class="fup">(' +
product;
            fact3 += ')</span><span class="fdn">(' + factor1 + ')
</span></div> = ' + factor2;
            fact4 = '<div class="fraction"><span class="fup">(' +
product;
            fact4 += ')</span><span class="fdn">(' + factor2 + ')
</span></div> = ' + factor1;
        }
    }
    else if (type==='fraction') {
        if (operation==='addSub') {
            fact1 = term1 + ' + ' + term2 + ' = ' + sum;
            fact2 = term2 + ' + ' + term1 + ' = ' + sum;
            fact3 = sum + ' &ndash; ' + term1 + ' = ' + term2;
            fact4 = sum + ' &ndash; ' + term2 + ' = ' + term1;
        }
        else {
            fact1 = factor1 + ' &sdot; ' + factor2 + ' = ' + product;
            fact2 = factor2 + ' &sdot; ' + factor1 + ' = ' + product;
            fact3 = product + ' &divide; ' + factor1 + ' = ' +
factor2;
            fact4 = product + ' &divide; ' + factor2 + ' = ' +
factor1;
        }
    }
    else if (type==='decimal') {
        if (operation==='addSub') {
            fact1 = term1 + ' + ' + term2 + ' = ' + sum;
            fact2 = term2 + ' + ' + term1 + ' = ' + sum;
            fact3 = sum + ' &ndash; ' + term1 + ' = ' + term2;
            fact4 = sum + ' &ndash; ' + term2 + ' = ' + term1;
        }
    }
} // end of getFacts()

// show the answer
```

```
function showAnswer() {
    answer = fact1 + '<br />' + fact2 + '<br />' + fact3 + '<br />' +
fact4;
    document.getElementById('answerSpace').innerHTML = answer;
} // end of showAnswer()

</script>



<body>

<head>
    <link rel='stylesheet' href='externalCSS/display fractions.css'>
</head>

Fact Families

<br /><br />

Which types of problems would you like?
<br /><input type='radio' id='wholeNumbersRadio' name='problem'
/>Whole Numbers
<br /><input type='radio' id='integersRadio' name='problem' />Integers
<br /><input type='radio' id='fractionsRadio' name='problem'
/>Fractions
<br /><input type='radio' id='decimalsRadio' name='problem' />Decimals
<br /><input type='radio' id='oneStepRadio' name='problem' />One-step
equations (single variable)
<br /><input type='radio' id='linearRadio' name='problem' />Linear
equations (two variables)
<br /><input type='radio' id='literalRadio' name='problem' />Literal
equations (three variables)
<br /><br />

<div id='errorMessageDiv' style='color:red; visibility:hidden;'>Click
or tap on one of the options above</div>
<br /><br />

<input type='button' value='Get new problem'
onclick='getNewProblem();' />
<br /><br />

<div id='problemSpace'></div>
<br /><br />

<input type='button' value='Show answer' onclick='showAnswer();' />
<br /><br />

<div id='answerSpace'>This is the answer space</div>


</body>
```