

Arcade Expressions for the GISCorps COVID-19 Testing Site Locator App

Updated 1/23/2021

[Pop-up Title \(updated 7/25\)](#)

[Pop-up Custom Attribute Display \(updated 12/30\)](#)

[Expression 0: Status Color](#)

[Expression 1: Has Comments](#)

[Expression 2: Stipulations \(updated 3/13/21\)](#)

[Expression 3: Click here for directions](#)

[Expression 4: Services Offered and Municipality](#)

[Expression 5: Hours of Operation](#)

[Expression 6: TestTypeNote](#)

[Expression 7: TestTypeLongForm \(updated 7/25\)](#)

[Expression 8: TestTypeURL \(updated 8/12\)](#)

[Expression 9: Contact \(added 5/29\)](#)

[Expression 10: Phone number \(added 11/30\)](#)

[Expression 11: Vaccine Info Link \(added 12/30\)](#)

[Expression 12: Offers Vaccine \(added 12/30\)](#)

[Testing Site Symbols](#)

Pop-up Title (updated 7/25)

[//Updated to include the link to the testing site website](#)

```
<table style='border-spacing: 0px; width: 100%; table-layout: fixed'>
  <tbody>
```

```
    <tr>
```

```
      <td style='width: 24px'><span style='display: block; background:
{expression/expr0}; border-radius: 50%; width: 15px; height: 15px;'></span></td>
```

```
  |
```

Pop-up Custom Attribute Display (updated 12/30)

// Updated to include new Contacts expression to accommodate testing sites that don't have
 // a phone number.
 // Updated to reorder the contents (address, website, etc. closer to the top)
 // Updated to make "For more information, visit the testing site provider's website" bold
 // Updated to add new phone number expression
 // Updated to add vaccination expressions

```

<div style="text-align: center; color: rgb(0, 0, 0); max-width: 600px; margin-left: auto;
margin-right: auto;">
<table cellpadding="0" style="font-size: medium; background: {expression/expr0}; color:
rgb(255, 255, 255); border-collapse: collapse; border-spacing: 0px; width: 100%; table-layout:
fixed; margin: 0px -1px;">
  <tbody>
    <tr>
      <td style="padding: 8px 0px; padding-bottom: 10px"><b>{status}<br
/></b>{expression/expr5}<b><br />-----</b><br />{expression/expr2}</td>
    </tr>
  </tbody>
</table>
<font style=""><font size="3">

```

```

Test type: </font><span style="color: rgb(76, 76, 76);"><font
size="3">{expression/expr7}</font><br /><font size="2"> </font></span></font><font
size="2">{expression/expr6} <a href="{expression/expr8}" rel="nofollow ugc" style=""
target="_blank">Learn More</a></font><br /></div><div style="text-align: center; color: rgb(0, 0,
0); max-width: 600px; margin-left: auto; margin-right: auto;"><font size="2" style=""><font
size="3"><br /></font></font><font size="3">{expression/expr4} <br /><br /></font><span
style="font-size: medium;">{fulladdr}</span><font size="3"> </font><br /><font size="3"
style="">

```


For the most up-to-date information, visit the testing site provider's website

<{comments}>

<{expression/expr12}<{expression/expr11}

Last update: {EditDate}.
For the most current and authoritative information about COVID-19 testing in your area, visit your local health department website.

Please verify this information by contacting the testing center, as specific details may have changed. If you are experiencing extreme or dangerous symptoms (including trouble breathing), please seek medical attention immediately.

Suggest an update

</div>

Expression 0: Status Color

// This expression determines the color of the block at the top and of the point in the popup title

```
if ($feature.status == 'Open') {  
    return '#83C96E'  
}  
if ($feature.status == 'Scheduled to Open') {  
    return '#C2C5CC'  
}  
if ($feature.status == 'Operational') {  
    return '#83C96E'  
}  
if ($feature.status == 'Closed') {  
    return '#C93100'  
}  
if ($feature.status == 'Temporarily Closed') {  
    return '#FF6600'  
}  
if ($feature.status == 'Scheduled to Close') {  
    return '#FFA500'  
}  
if ($feature.status == 'Permanently Closed') {  
    return '#C93100'  
}  
if ($feature.status == 'Impacted') {  
    return '#007AC2'  
}  
if ($feature.status == 'Testing Restricted') {  
    return '#7c7979'  
}  
return null;
```

Expression 1: Has Comments

// This expression returns the comments text only if there is a comment.

```
!If(IsEmpty($feature.comments), "none", null);
```

Expression 2: Stipulations (updated 3/13/21)

```
// This expression returns the additional text that appears in the block of color
// at the top of the popup.
// Edited to note when a vehicle is required.
// Edited to add "call first" note
// Edited to add take-home test stipulation.
// Edited to add "offers vaccine" stipulation
// Updated appointments section to include two more options (Test only, Vax only)
// Updated drive-through and referral sections to include two more options (Test only, Vax only)
// Updated to add carriage returns instead of semicolons separating items
```

```
// Write a script to return a value to show in the pop-up.
// For example, get the average of 4 fields:
// Average($feature.SalesQ1, $feature.SalesQ2, $feature.SalesQ3, $feature.SalesQ4)
var requirements = "";
var comma = "";

if ($feature["referral_required"] == 'Yes') {
    requirements = 'Physician referral required'
    comma = '\x0d\x0a'
} else if ($feature["referral_required"] == 'No') {
    requirements = 'No physician referral required'
    comma = '\x0d\x0a'
} else if ($feature["referral_required"] == 'Test only') {
    requirements = 'No physician referral required for testing'
    comma = '\x0d\x0a'
} else if ($feature["referral_required"] == 'Vax only') {
    requirements = 'No physician referral required for vaccination'
    comma = '\x0d\x0a'
}

if ($feature["appt_only"] == 'Yes') {
    requirements = concatenate([requirements, 'Appointment required'], comma)
    comma = "\x0d\x0a"
} else if ($feature["appt_only"] == 'No') {
    requirements = concatenate([requirements, 'No appointment required'], comma)
    comma = "\x0d\x0a"
} else if ($feature["appt_only"] == 'Test only') {
    requirements = concatenate([requirements, 'Appointment required for testing'], comma)
    comma = "\x0d\x0a"
} else if ($feature["appt_only"] == 'Vax only') {
```

```

requirements = concatenate([requirements, 'Appointment required for vaccination'], comma)
comma = "\x0d\x0a"
}
if ($feature["drive_through"] == 'Yes') {
    requirements = concatenate([requirements, 'Drive-through'], comma)
    comma = "\x0d\x0a"
} else if ($feature["drive_through"] == 'No') {
    requirements = concatenate([requirements, 'No drive-through'], comma)
    comma = "\x0d\x0a"
} else if ($feature["drive_through"] == 'Test only') {
    requirements = concatenate([requirements, 'Drive-through for testing only'], comma)
    comma = "\x0d\x0a"
} else if ($feature["drive_through"] == 'Vax only') {
    requirements = concatenate([requirements, 'Drive-through for vaccination only'], comma)
    comma = "\x0d\x0a"
}
if ($feature["call_first"] == 'Yes') {
    requirements = concatenate([requirements, 'Call first'], comma)
    comma = "\x0d\x0a"
}
if ($feature["take_home_test"] == 'Yes') {
    requirements = concatenate([requirements, 'Take-home test'], comma)
    comma = "\x0d\x0a"
}
if ($feature["offers_vaccine"] == 'Yes') {
    requirements = concatenate([requirements, 'Offers COVID-19 vaccination'], comma)
    comma = "\x0d\x0a"
}
requirements

```

Expression 3: Click here for directions

```

// This expression returns the "Click here for directions" text only if there is text
// in the address field
// This is not currently being used because we're using the Esri geocoder now.
// It was used when we were sending people to Google Maps to avoid consuming credits.

```

```

if (IsEmpty($feature.fulladdr)==false) {
    return 'Click here for directions.'
}

```

Expression 4: Services Offered and Municipality

// This expression constructs the text that appears beneath the status in the block of color in the popup. Example: "COVID-19 screening only (no testing) in Ennis"

```

If (IsEmpty($feature.municipality)==false) {
    if (IsEmpty($feature["services_offered_onsite"])==false) {
        return Concatenate([DomainName($feature,"services_offered_onsite"), ' in ',
$feature.municipality], ")")
    } else {
        return $feature.municipality
    }
} else {
    if (IsEmpty($feature["services_offered_onsite"])==false) {
        return DomainName($feature,"services_offered_onsite")
    } else {
        return "
    }
}

```

Expression 5: Hours of Operation

// This expression returns the hours of operation text only if that field has data.

```

if (IsEmpty($feature.operhours)) {
    return "
} else {
    return concatenate(['Hours: ', $feature.operhours], ")
}

```

Expression 6: TestTypeNote

```
// This expression adds a warning about antibody tests if the test type includes
// the word "antibody" (the 'a' is left off deliberately to avoid issues of case).
```

```
If (find("ntibody",$feature["type_of_test"],0)>0) {
    return "Antibody tests are not FDA approved for diagnosis of COVID-19."
}
```

Expression 7: TestTypeLongForm (updated 7/25)

```
// This expression translates the Test Type domain values into more readable versions.
// Updated to mirror CDC test type verbiage (viral and antibody)
// Updated to include "all three" (molecular, antigen, and antibody)
```

```
// Write a script to return a value to show in the pop-up.
// For example, get the average of 4 fields:
// Average($feature.SalesQ1, $feature.SalesQ2, $feature.SalesQ3, $feature.SalesQ4)
```

```
if ($feature["type_of_test"]=="molecular") {
    return "Molecular Test"
} else if ($feature["type_of_test"]=="antibody") {
    return "Antibody Test"
} else if ($feature["type_of_test"]=="molecular and antibody") {
    return "Both Molecular and Antibody Tests"
} else if ($feature["type_of_test"]=="molecular and antigen") {
    return "Both Molecular and Antigen Tests"
} else if ($feature["type_of_test"]=="antibody and antigen") {
    return "Both Antibody and Antigen Tests"
} else if ($feature["type_of_test"]=="all three") {
    return "Molecular, Antigen, and Antibody Tests"
} else if ($feature["type_of_test"]=="none") {
    return "None (screening only)"
} else if ($feature["type_of_test"]=="not specified") {
    return "Not specified"
} else if ($feature["type_of_test"]=="needs more research") {
    return "Not specified"
} else if (isEmpty($feature["type_of_test"])) {
    return "Not specified"
}
```

Expression 8: TestTypeURL (updated 8/12/20)

```
// This expression provides an appropriate URL for the "Learn More" link under Test Type.  
// NOTE: you should replace this URL as better public explanations become available  
// from CDC or FDA  
// Updated to use new CDC test type webpage  
// Updated to use new FDC webpage that covers all three test types
```

```
return  
"https://www.fda.gov/consumers/consumer-updates/coronavirus-testing-basics#:~:text=Types%20of%20Tests,-There%20are%20two&text=Currently%20there%20are%20two%20types,the%20surface%20of%20the%20virus."
```

Expression 9: Contact (added 5/29/20)

```
// Added to accommodate testing sites without phone numbers.  
// This expression writes one sentence if there is a phone number and another if there  
// is not a phone number available.
```

```
If (isEmpty($feature.phone)) {  
    return "For more information, please visit the testing site provider's"  
} else if (left($feature.phone, 1) != "n") {  
    return Concatenate(["For more information, please call ", $feature.phone, " or visit the testing  
provider's"], "")  
}
```

Expression 10: Phone number (added 11/30/20)

```
//Added to avoid displaying an empty space where phone numbers are missing.
```

```
If(IsEmpty($feature.phone), ".", CONCATENATE([" or call ", ($feature.phone), ":", ""]));
```

Expression 11: Vaccine Info Link (added 12/30)

// Added to include a URL for vaccine-specific information when vaccines are offered at a site

```
If ($feature["offers_vaccine"]=='Yes') {  
  IF (FIND('http', $feature["vaccine_url"], 0) > -1) {  
    Return 'Learn more.'  
  }  
}
```

Expression 12: Offers Vaccine (added 12/30)

// Added to include a note that vaccination is offered at a site.

```
If ($feature["offers_vaccine"]=='Yes') {  
  Return 'This site offers COVID-19 vaccination. '  
}
```

Testing Site Symbols

