



# Introduction to Tax-Cruncher

Matt Jensen and Peter Metz

January 2020

## Key Points

- Tax-Cruncher is a new open-source project for analyzing how tax policy affects taxpayer finances and incentives. Tax-Cruncher allows policymakers, researchers, and everyday taxpayers to analyze how changes to current tax law would affect a taxpayer's financial situation.
- To leverage Tax-Cruncher's capabilities, users choose between an easy-to-use web application and a Python application programming interface for the entry point that best fits their work and skills.
- Using either entry point, users can calculate federal individual income and payroll tax liabilities under current law for any year between 2013 and 2028, design tax reforms using over 200 parameterized features of the US tax code, and compare current law and policy reforms based on a broad set of metrics including changes in tax liabilities, credits, and average and marginal rates.

---

Tax-Cruncher is open-source software for users to estimate how policy reforms or alternative income scenarios could influence a taxpayer's tax liabilities and incentives. Users specify two types of inputs: a taxpayer's financial and demographic information necessary for computing federal tax liabilities and, optionally, a policy reform to the tax code. Tax-Cruncher generates tables and charts that display a detailed analysis of the taxpayer's federal tax liabilities and marginal tax rates, as well as how those liabilities and rates change under new policy scenarios and alternative taxpayer earnings.

Users can choose from two interfaces: an easy-to-use web application for generating preset tables and charts<sup>1</sup> or a Python application programming interface (API), which affords researchers and analysts broader

capabilities.<sup>2</sup> The Tax-Cruncher project seeks to be transparent, community centric, and helpful to outside researchers. The source code for Tax-Cruncher<sup>3</sup> and the other modeling projects Tax-Cruncher depends on<sup>4</sup> are open source and hosted on GitHub, and Tax-Cruncher is cross-compatible with other projects in the Policy Simulation Library.<sup>5</sup> Users are encouraged to contribute to the project and welcome to modify the source code to fit their needs.

Tax-Cruncher can accommodate information from many but not all types of taxpayers. Important taxpayer information relevant to some returns is not considered by Tax-Cruncher. If you find a bug, please report it to the authors. Tax-Cruncher should not be used for filing taxes.

## Case Study: Temporary Provisions of the Tax Cuts and Jobs Act

The Tax Cuts and Jobs Act (TCJA), signed into law in December 2017, featured a number of significant individual income tax reforms set to expire at the end of 2025.<sup>6</sup> If no new law is passed by 2026, the temporary provisions will revert back to their values for 2026 as calculated according to pre-TCJA law but using TCJA's new inflation adjustments.

The case study proceeds in two parts. First, we use Tax-Cruncher to analyze how the TCJA extension would affect a hypothetical middle-class family, filing jointly, in 2026. In this case study, we focus on the five significant temporary changes relevant to this taxpayer (Tables 1–5).

**Table 1. New Personal Income Tax Rates and Brackets for Joint Filers**

Pre-TCJA		TCJA	
Rate	Threshold (2017)	Rate	Threshold (2018)
10%	\$18,650	10%	\$19,050
15%	\$75,900	12%	\$77,400
25%	\$153,100	22%	\$165,000
28%	\$233,350	24%	\$315,000
33%	\$416,700	32%	\$400,000
35%	\$470,700	35%	\$600,000
39.6%	\$470,701+	37%	\$600,001+

Source: Internal Revenue Service, *Tax Reform: Basics for Individuals and Families*, 2018, <https://www.irs.gov/pub/irs-pdf/p5307.pdf>.

**Table 2. Increased Standard Deduction for Joint Filers**

Pre-TCJA (2017)	TCJA (2018)
\$12,700	\$24,000

Source: Internal Revenue Service, *Tax Reform: Basics for Individuals and Families*, 2018, <https://www.irs.gov/pub/irs-pdf/p5307.pdf>.

**Table 3. Repealed Personal Exemptions for Joint Filers**

Pre-TCJA (2017)	TCJA (2018)
\$8,100	\$0

Source: Internal Revenue Service, *Tax Reform: Basics for Individuals and Families*, 2018, <https://www.irs.gov/pub/irs-pdf/p5307.pdf>.

**Table 4. Increased Child Tax Credit for Joint Filers**

	Pre-TCJA (2017)	TCJA (2018)
Maximum Credit per Child	\$1,000	\$2,000
Refundable Portion	\$1,000	\$1,400
Income Threshold for Refundable Portion	\$3,000	\$2,500
Credit for Other Dependents	\$0	\$500
Phase Out Start	\$200,000	\$400,000

Source: Internal Revenue Service, *Tax Reform: Basics for Individuals and Families*, 2018, <https://www.irs.gov/pub/irs-pdf/p5307.pdf>.

**Table 5. Ceiling on the Amount of Deductible State, Local, and Property Taxes for Joint Filers**

Pre-TCJA (2017)	TCJA (2018)
None	\$10,000

Source: Internal Revenue Service, *Tax Reform: Basics for Individuals and Families*, 2018, <https://www.irs.gov/pub/irs-pdf/p5307.pdf>.

In Part I of our case study, we discuss web application inputs for both the taxpayer and the tax policy reform and walk through the interactive outputs. In Part II, using Tax-Cruncher's Python API, we analyze how the TCJA extension would affect a batch of filing units. We discuss the inputs that the API accepts and the code we use to create an output table. While basic programming abilities are helpful to use the API, users can take advantage of its capabilities with only a few lines of code.

## Part I: Using the Tax-Cruncher Web Application

We will use the Tax-Cruncher web application to explore how TCJA's extension would affect a married filing unit with two children. The taxpayer has \$90,000 of primary earner wages and \$20,000 of deductible expenses, comprised of \$10,000 in property taxes and \$10,000 in home mortgage interest. We will examine how this taxpayer fares in 2026 under current law (i.e., with tax provisions reverting back to their pre-TCJA values) and an alternative policy regime in which the expiring provisions are extended.

**User Inputs.**<sup>7</sup> Tax-Cruncher presents two sets of inputs: “Tax Information” and “Policy.” The first set of inputs accepts the financial and demographic data necessary to compute federal tax liabilities. In our case study, we enter the “Tax Information” inputs to reflect the middle-class filer described above. For example, in Figure 1, we enter the filer’s marital status and age. We fill out similar input boxes to enter our filer’s dependents, income, and expenses, which you can find step-by-step in Appendix A.

The second set of inputs, “Policy,” contains many of the US provisions for individual income and payroll taxes and some provisions not enacted under the current US federal tax code, such as a universal basic income. The default values reflect the effective law for the year selected according to current law.<sup>8</sup> To adjust a default value, we simply enter a new value, and the text box will turn blue. Throughout the app, we may fill in as many inputs as we find useful. For example, if we did not modify the “Policy” section, Tax-Cruncher would analyze our filer’s liabilities under current law, without a policy reform.

In our case study, we modify the “Policy” inputs to extend the five major temporary provisions of the TCJA outlined above. For example, Figure 2 shows how we repeal the personal exemption,<sup>9</sup> and Figure 3 adjusts the parameters used to calculate the nonrefundable portion of the child tax credit (CTC)<sup>10</sup> to align with TCJA values. The values entered into the blue boxes are the tax reforms we implement, and the white boxes are the default values that we did not modify.

Figure 1. Demographics

The screenshot shows the 'Demographics' section of the Tax-Cruncher application. It contains three input fields: 'Marital Status' with a dropdown menu set to 'Joint' (default is 'Single'), 'Age of primary taxpayer December 31st of tax year (or 0)' with a text box containing '40' (default is '0'), and 'Age of spouse' with a text box containing '40' (default is '0'). Each field has a blue border and a small information icon.

Source: Compute Studio, “PSLmodels/Tax Cruncher,” <https://compute.studio/PSLmodels/Tax-Cruncher/291/edit/>.

Figure 2. Personal Exemptions

The screenshot shows the 'Personal Exemptions' section of the Tax-Cruncher application. It contains three input fields: 'Personal And Dependent Exemption Amount' with a text box containing '0' (default is '4901'), 'Personal Exemption Phaseout Rate' with a text box containing '0.02' (default is '0.02'), and 'Repeal for Dependents Under Age 18' with a text box containing 'false' (default is 'false'). Each field has a blue border and a small information icon.

Source: Compute Studio, “PSLmodels/Tax Cruncher,” <https://compute.studio/PSLmodels/Tax-Cruncher/291/edit/>.

**Tax-Cruncher Outputs.**<sup>11</sup> When the simulation is finished, Tax-Cruncher displays a series of tables and charts. These outputs offer an overview of the taxpayer’s tax situation under both current law and the reform. They also give the user an idea of how the taxpayer would fare with alternative income arrangements, holding other filing unit characteristics constant.

Figure 4 is the “Basic Liabilities” table, in which we can examine individual income and payroll tax liabilities and rates and how those liabilities and rates change from the user-specified policy reform

**Figure 3. Child or Dependent Credits**

Child Tax Credit

Maximum nonrefundable child tax credit per child 1 CPI

2000

Default: 1000

Bonus child tax credit maximum for qualifying children under five 1 CPI

0

Child tax credit phaseout MAGI start 1 CPI

single

200000

Default: 75000

mjoint

400000

Default: 110000

mseparate

200000

Default: 55000

headhh

200000

Default: 75000

Child and dependent tax credit phaseout rate 1

0.05

Source: Compute Studio, “PSLmodels/Tax Cruncher,” <https://compute.studio/PSLmodels/Tax-Cruncher/291/edit/>.

**Figure 4. Basic Liabilities**

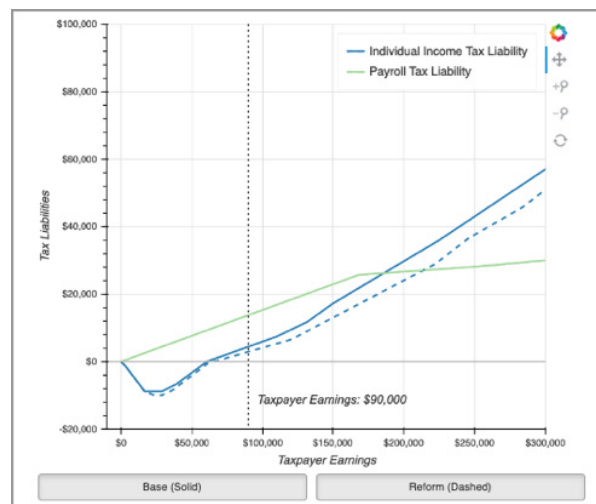
	Base	Reform	Change
Individual Income Tax	4430.95	2926.11	-1504.84
Income Tax Marginal Rate	0.14	0.11	-0.03
Employee + Employer Payroll Tax	13770.00	13770.00	0.00
Payroll Tax Marginal Rate	0.14	0.14	0.00

Source: Compute Studio, “PSLmodels/Tax Cruncher,” <https://compute.studio/PSLmodels/Tax-Cruncher/291/edit/>.

scenario. Figure 4 shows that the TCJA extension lowers the tax liability of our middle-class household in 2026 by \$1,505.

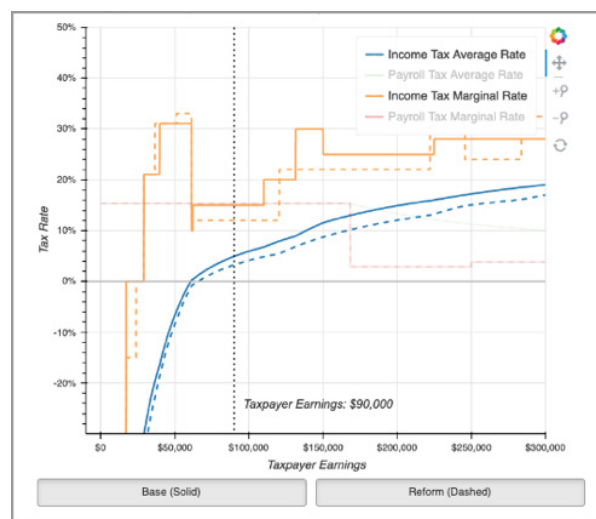
Next, Tax-Cruncher displays three interactive skyline charts to analyze tax liabilities, marginal and average tax rates, and tax credits across a range of incomes. The figures are created by incrementing primary earner wages (or a different income measure of your choosing) from \$0 to \$500,000 (holding other taxpayer characteristics constant), and they allow us to visualize the effects of current tax law and reform on the taxpayer over a range of wages. The first output of our case study, “Tax Liabilities by Wage,” in Figure 5, shows individual income tax and payroll tax liabilities by wage under the current law (solid) and the reform policy (dashed). As taxpayer wage

**Figure 5. Tax Liabilities by Wage (Holding Other Inputs Constant)**



Source: Compute Studio, “PSLmodels/Tax Cruncher,” <https://compute.studio/PSLmodels/Tax-Cruncher/291/edit/>.

**Figure 6. Tax Rates by Wage (Holding Other Inputs Constant)**

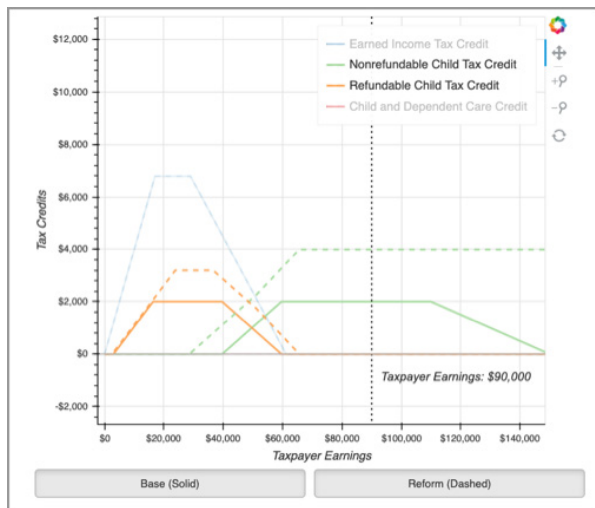


Source: Compute Studio, “PSLmodels/Tax Cruncher,” <https://compute.studio/PSLmodels/Tax-Cruncher/291/edit/>.

increases, so does the dollar value of the extension, as represented by the increasing gap between the “Base” (i.e., current law) and “Reform” lines in this figure.

The second output chart created by Tax-Cruncher, “Tax Rates by Wage,” shows the average and marginal tax rates of the income and payroll taxes by wage. In Figure 6, we present the lines for the average and

**Figure 7. Tax Credits by Wage (Holding Other Inputs Constant)**



Source: Compute Studio, “PSLmodels/Tax Cruncher,” <https://compute.studio/PSLmodels/Tax-Cruncher/291/edit/>.

marginal income tax rates under the base policy (solid line) and TCJA extension (dashed line) in 2026. We observe that TCJA extension results in lower average tax rates across all wages for our taxpayer, while its effect on the marginal tax rate can be either positive or negative, dependent on the wage.

The final figure created by Tax-Cruncher, Figure 7, displays the amount of credits received from the refundable and nonrefundable portions of the CTC, the earned income tax credit, and the dependent care credit. Figure 7 presents the effect of TCJA’s expanded refundable and nonrefundable portions of the CTC, represented by the orange and green lines, respectively. For the nonrefundable portion of CTC, the maximum credit doubles, and the phaseout threshold increases. For the refundable portion, the maximum credit per dependent is raised to \$1,600, and the income threshold is lowered to \$2,500.<sup>12</sup>

Finally, Tax-Cruncher calculates and displays all the components of individual income tax in a table called “Calculation of Liabilities,” presented in Figure 8. The first column displays 2026 law components, the second shows the policy reform, and the third adds one dollar to the primary taxpayer’s wages, which is useful for marginal tax rate analysis. Figure 8 gives concrete numbers that make it possible to

**Figure 8. Calculation of Liabilities**

	Base	Reform	+ \$1 (Taxpayer Earnings)
Adjusted Gross Income (AGI)	90000.00	90000.00	90001.00
Unemployment Insurance in AGI	0.00	0.00	0.00
Social Security in AGI	0.00	0.00	0.00
Itemized Deductions	20000.00	0.00	0.00
Taxable Income	50396.00	61489.34	61490.34
Child Tax Credit (CTC)	2000.00	4000.00	4000.00
CTC Refundable	0.00	0.00	0.00
Child Care Credit	0.00	0.00	0.00
Earned Income Tax Credit (EITC)	0.00	0.00	0.00
Alternative Minimum Tax (AMT) Taxable Income	80000.00	90000.00	90001.00
AMT Liability	0.00	0.00	0.00
Net Investment Income Tax	0.00	0.00	0.00
Income Tax Before Credits	6430.95	6926.11	6926.23

Source: Compute Studio, “PSLmodels/Tax Cruncher,” <https://compute.studio/PSLmodels/Tax-Cruncher/291/edit/>.

understand how the various provisions of the tax law and tax reform interact.

First, in the row labeled “Itemized Deductions,” we see that our taxpayer itemizes their deductions (\$20,000) under 2026 law but takes the standard deduction (\$28,510) under the TCJA extension. Even so, looking at the row labeled “Taxable Income,” we see that the taxpayer’s taxable income is almost \$11,000 greater under the TCJA extension because the elimination of the personal exemption outweighs the effect of the greater standard deduction.<sup>13</sup> Finally, the CTC doubles for our taxpayer under the TCJA extension. While regular tax is greater under the TCJA extension, our filer sees an overall decrease in liabilities because of the expanded CTC.

## Part II: Using the Tax-Cruncher Python API

We have seen how we can use Tax-Cruncher to analyze the effects of a TCJA extension on a hypothetical middle-class family. Now let us explore what the Python API can do.<sup>14</sup> While the web application requires no programming knowledge and can create



interactive charts in less than a minute, the API has a few advantages over the web application. We can:

- Analyze multiple households simultaneously,
- Compare the impact of two reforms by setting the baseline policy to a reform, and
- Customize outputs by modifying the source code.

The first step to using the API is to create a CSV input file. Each column contains data on a different tax information variable (e.g., year, marital status, wages, etc.), and each row represents a different filing unit.<sup>15</sup> Figure 9 provides a snapshot of our input data.

Then, we create a JSON file that specifies the provisions of the tax reform we want to analyze. The JSON file's structure adheres to the same rules as tax policy reform files for Tax-Calculator.<sup>16</sup> Figure 10 shows the section of the JSON file that modifies personal income brackets and rates to the values specified in the TCJA extension. Tax-Calculator will extend specified parameters to future years and, for parameters such as tax brackets, will adjust the values for inflation.

Once we have gathered our input data and specified our policy reform JSON file, we can analyze the TCJA extension's impact on each filer in our input file with just a few lines of Python code. The Python API allows us to create two off-the-shelf tables:

1. A table that shows liabilities, credits, rates, and more, under either the baseline policy or the reform policy, and
2. A table that shows the differences between the reform and baseline policies.

In just a few lines of code—shown in Figure 11—we generate a table that displays the liability, for each filer in the input file, under 2026 current law.

**Figure 9. Python API Input Data**

	A	B	C	D	E	F	G	H	I	J
1	1	2026	1	40	0	0	0	0	0	10000
2	2	2026	1	40	0	0	0	0	0	30000
3	3	2026	1	40	0	0	0	0	0	60000
4	4	2026	1	40	0	0	0	0	0	100000
5	5	2026	1	40	0	0	0	0	0	200000
6	6	2026	1	40	0	0	0	0	0	500000
7	7	2026	1	40	0	1	1	1	1	10000
8	8	2026	1	40	0	1	1	1	1	30000
9	9	2026	1	40	0	1	1	1	1	60000
10	10	2026	1	40	0	1	1	1	1	100000
11	11	2026	1	40	0	1	1	1	1	200000
12	12	2026	1	40	0	1	1	1	1	500000

Source: GitHub, "PSLmodels / Tax-Cruncher," [https://github.com/PSLmodels/Tax-Cruncher/blob/master/docs/WP\\_example/input.csv](https://github.com/PSLmodels/Tax-Cruncher/blob/master/docs/WP_example/input.csv).

**Figure 10. Python API Tax Reform JSON File**

```
"II_rt1": {"2018": 0.10},
"II_rt2": {"2018": 0.12},
"II_rt3": {"2018": 0.22},
"II_rt4": {"2018": 0.24},
"II_rt5": {"2018": 0.32},
"II_rt6": {"2018": 0.35},
"II_rt7": {"2018": 0.37},
"II_brk1": {"2018": [9525, 19050, 9525, 13600, 19050]},
"II_brk2": {"2018": [38700, 77400, 38700, 51800, 77400]},
"II_brk3": {"2018": [82500, 165000, 82500, 82500, 165000]},
"II_brk4": {"2018": [157500, 315000, 157500, 157500, 315000]},
"II_brk5": {"2018": [200000, 400000, 200000, 200000, 400000]},
"II_brk6": {"2018": [500000, 600000, 300000, 500000, 600000]},
```

Source: GitHub, "PSLmodels / Tax-Cruncher," [https://github.com/PSLmodels/Tax-Cruncher/blob/master/docs/WP\\_example/input.csv](https://github.com/PSLmodels/Tax-Cruncher/blob/master/docs/WP_example/input.csv).

**Figure 11. Calling the Python API**

```
In [1]: from taxcrunch import multi_cruncher as mc
        batch = mc.Batch('../docs/WP_example/input.csv')
        output_df = batch.create_table(reform_file='../docs/WP_example/TCJA_ext.json')
        output_df

Out[1]:
```

	ID	Individual Income Tax	Payroll Tax	Wages	AGI	UI in AGI	OASDI in AGI	Itemized Deductions	Taxable Inc	Regular Tax
0	1.0	-616.550000	1530.0000	10000.0	10000.0	0.0	0.0	0.0	0.00	0.0000
1	2.0	1663.055800	4590.0000	30000.0	30000.0	0.0	0.0	0.0	15744.66	1663.0558
2	3.0	5263.055800	9180.0000	60000.0	60000.0	0.0	0.0	0.0	45744.66	5263.0558
3	4.0	14040.175800	15300.0000	100000.0	100000.0	0.0	0.0	0.0	85744.66	14040.1758
4	5.0	37794.960400	26674.0112	200000.0	200000.0	0.0	0.0	0.0	185744.66	37794.9604
5	6.0	141131.099700	38074.0112	500000.0	500000.0	0.0	0.0	0.0	485744.66	141131.0997
6	7.0	-4525.000000	1530.0000	10000.0	10000.0	0.0	0.0	0.0	0.00	0.0000
7	8.0	-3998.072992	4590.0000	30000.0	30000.0	0.0	0.0	0.0	8617.00	861.7000
8	9.0	2310.919000	9180.0000	60000.0	60000.0	0.0	0.0	0.0	38617.00	4310.9190
9	10.0	8819.066000	15300.0000	100000.0	100000.0	0.0	0.0	0.0	78617.00	10819.0660
10	11.0	32431.297400	26674.0112	200000.0	200000.0	0.0	0.0	0.0	178617.00	34431.2974
11	12.0	136983.394100	38074.0112	500000.0	500000.0	0.0	0.0	0.0	478617.00	136983.3941
12	13.0	-616.550000	1530.0000	10000.0	10000.0	0.0	0.0	0.0	0.00	0.0000

Source: GitHub, "PSLmodels / Tax-Cruncher," [https://github.com/PSLmodels/Tax-Cruncher/blob/master/docs/WP\\_example/TCJA\\_ext\\_analysis.ipynb](https://github.com/PSLmodels/Tax-Cruncher/blob/master/docs/WP_example/TCJA_ext_analysis.ipynb).

## A Gateway to Open-Source Models

Tax-Cruncher aims to inform voters and offer researchers a tool for in-depth and objective policy analysis. In addition to Tax-Cruncher's professional and academic uses, the web application allows taxpayers with no programming experience to harness its capabilities. Students can study how the various provisions of tax law interact, financial planners can estimate their clients' tax burdens in future years and under different tax laws, and taxpayers can understand how their taxes are calculated and how they might change in the future.

Tax-Cruncher also provides a stepping-stone for new policy analysts and programmers to use and contribute to public policy models in the open-source ecosystem. After mastering Tax-Cruncher, new users can explore other models incubated by the Open Source Policy Center<sup>17</sup> and cataloged by the Policy Simulation Library<sup>18</sup> for a broader understanding of policy. For example, Tax-Calculator<sup>19</sup> provides revenue estimates and distributional effects of individual income and payroll tax law, and Cost-of-Capital-Calculator<sup>20</sup> evaluates tax policy's effect on businesses' investment incentives. With Tax-Cruncher, these models offer a powerful suite to inform the tax policy debate and generate better tax policy.

## About the Authors

**Matt Jensen** is the director of the Open Source Policy Center at the American Enterprise Institute and a maintainer of the Tax-Calculator project, on which Tax-Cruncher depends.

**Peter Metz** is a research associate at the Open Source Policy Center and a maintainer of the Tax-Cruncher project. He may be reached at [peter.metz@aei.org](mailto:peter.metz@aei.org).

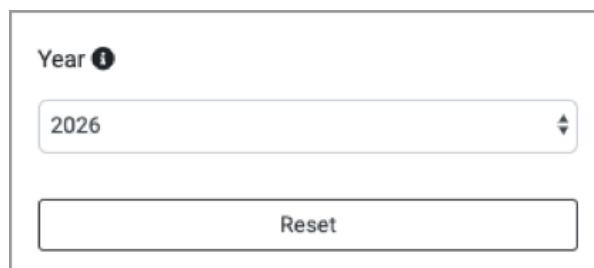
## Appendix A

In the following section, we show the complete set of inputs used to generate the case study simulation. We start by selecting 2026 as the year for the case study. By selecting 2026 and pressing the “Reset” button, the policy parameter values are automatically set to their 2026 current law values (i.e., after the TCJA’s temporary provisions expire).

Then, we input the characteristics of the tax unit we want to analyze. The “Tax Information” section of inputs is comprised of four subsections: “Demographics,” “Dependents,” “Income,” and “Taxes and Expenses.” Note that for the “Income” subsection, the fields not applicable to our case study’s filing unit are omitted in Figure A2 (i.e., various types of capital income).

Once we complete the “Tax Information” section of inputs, we proceed to the “Policy” section of inputs to specify the tax policy reform we want to analyze. In our case, we modify the parameters that correspond to the five temporary provisions of the TJCA expansion.

**Figure A1. Set Parameters to 2026 Current Law Values**

The image shows a user interface element for selecting a year. It features a label 'Year' with an information icon. Below the label is a dropdown menu with '2026' selected. At the bottom of the dropdown is a 'Reset' button.

Source: Compute Studio, “PSLmodels/Tax Cruncher,” <https://compute.studio/PSLmodels/Tax-Cruncher/291/edit/>.

1. Repeal the personal exemption (Figure A3).
2. Increase the standard deduction (Figure A4).
3. Cap the amount of deductible state, local, and property taxes (Figure A5).
4. Increase the child tax credit (Figure A6).
5. Modify personal income tax rates and brackets (Figure A7).<sup>21</sup>



**Figure A2. Tax Information Inputs**

Demographics	Income
<p><b>Marital Status</b> ⓘ</p> <p>Joint</p> <p>Default: Single</p>	<p><b>Wage and salary income of primary taxpayer</b> ⓘ</p> <p>90000</p> <p>Default: 0</p>
<p><b>Age of primary taxpayer December 31st of tax year (or 0)</b> ⓘ</p> <p>40</p> <p>Default: 0</p>	<p><b>Wage and salary income of spouse</b> ⓘ</p> <p>0</p> <p>Default: 0</p>
<p><b>Age of spouse</b></p> <p>40</p> <p>Default: 0</p>	
Dependents	Taxes and Expenses
<p><b>Number of dependents</b> ⓘ</p> <p>2</p> <p>Default: 0</p>	<p><b>Itemizable real estate taxes paid</b></p> <p>10000</p> <p>Default: 0</p>
<p><b>Number of children under 13 with eligible child care expenses</b> ⓘ</p> <p>2</p> <p>Default: 0</p>	<p><b>Itemizable state and local income/sales taxes</b></p> <p>0</p>
<p><b>Number of children under 17 for the entire tax year</b> ⓘ</p> <p>2</p> <p>Default: 0</p>	<p><b>Child care expenses</b> ⓘ</p> <p>0</p>
<p><b>Number of qualifying children for EITC</b> ⓘ</p> <p>2</p> <p>Default: 0</p>	<p><b>Other itemizable expenses</b> ⓘ</p> <p>10000</p> <p>Default: 0</p>

Source: Compute Studio, "PSLmodels/Tax Cruncher," <https://compute.studio/PSLmodels/Tax-Cruncher/291/edit/>.

**Figure A3. Personal Exemptions**

**Personal Exemptions**

**Personal And Dependent Exemption Amount**

Personal and dependent exemption amount CPI

0

Default: 4901

Source: Compute Studio, "PSLmodels/Tax Cruncher," <https://compute.studio/PSLmodels/Tax-Cruncher/291/edit/>.

**Figure A4. Standard Deduction**

**Standard Deduction**

**Standard Deduction Amount**

Standard deduction amount CPI

single	mjoint	mseparate	headhh
14255.34	28510.66	14255.34	21383
Default: 7685	Default: 15369	Default: 7685	Default: 11315

Source: Compute Studio, "PSLmodels/Tax Cruncher," <https://compute.studio/PSLmodels/Tax-Cruncher/291/edit/>.

**Figure A5. Ceiling on State and Local Tax Deduction**

**Itemized Deductions**

**State And Local Taxes And Real Estate Taxes**

State and local income, sales, and real estate tax deduction haircut. CPI

0

Ceiling on the amount of state and local income, sales and real estate tax deductions allowed (dollars) CPI

single	mjoint	mseparate	headhh	widow
10000	10000	5000	10000	10000
Default: 9e+99	Default: 9e+99	Default: 9e+99	Default: 9e+99	Default: 9e+99

Source: Compute Studio, "PSLmodels/Tax Cruncher," <https://compute.studio/PSLmodels/Tax-Cruncher/291/edit/>.

**Figure A6. Child or Dependent Credits**

**Child/Dependent Credits**

**Child Tax Credit**

Maximum nonrefundable child tax credit per child CPI

2000

Default: 1000

Bonus child tax credit maximum for qualifying children under five CPI

0

Child tax credit phaseout MAGI start CPI

single	mjoint	mseparate	headhh	widow
200000	400000	200000	200000	400000
Default: 75000	Default: 110000	Default: 55000	Default: 75000	Default: 75000

Child and dependent tax credit phaseout rate CPI

0.05

**Additional Child Tax Credit**

Maximum refundable additional child tax credit CPI

1600

Default: 1000

Additional Child Tax Credit rate CPI

0.15

Bonus additional child tax credit rate for families with qualifying children under 5 CPI

0

Additional Child Tax Credit income threshold CPI

2500

Default: 3000

Additional Child Tax Credit minimum number of qualified children for different formula CPI

3

**Other Dependent Tax Credit**

Maximum nonrefundable other-dependent credit CPI

500

Default: 0

Source: Compute Studio, "PSLmodels/Tax Cruncher," <https://compute.studio/PSLmodels/Tax-Cruncher/291/edit/>.

**Figure A7. Personal Income Tax Rates and Brackets**

## Personal Income

Regular: Non-AMT, Non-Pass-Through

Personal income (regular/non-AMT/non-pass-through) tax rate 1

0.1

Personal income (regular/non-AMT/non-pass-through) tax bracket (upper threshold) 1

single	mjoint	mseparate	headhh	widow
11315.17	22630.36	11315.17	16156.05	22630.36
Default: 11285	Default: 22569	Default: 11285	Default: 16156	Default: 22569

Personal income (regular/non-AMT/non-pass-through) tax rate 2

0.12

Default: 0.15

Personal income (regular/non-AMT/non-pass-through) tax bracket (upper threshold) 2

single	mjoint	mseparate	headhh	widow
45973.46	91946.93	45973.46	61535.53	91946.93
Default: 45926	Default: 91851	Default: 45926	Default: 61476	Default: 91851

Personal income (regular/non-AMT/non-pass-through) tax rate 3

0.22

Default: 0.25

Personal income (regular/non-AMT/non-pass-through) tax bracket (upper threshold) 3

single	mjoint	mseparate	headhh	widow
98005.43	196010.9	98005.43	98005.43	196010.9
Default: 111214	Default: 185275	Default: 92638	Default: 158773	Default: 185275

Personal income (regular/non-AMT/non-pass-through) tax rate 4

0.24

Default: 0.28

Personal income (regular/non-AMT/non-pass-through) tax bracket (upper threshold) 4

single	mjoint	mseparate	headhh	widow
187101.31	374202.61	187101.31	187101.31	374202.61
Default: 231927	Default: 282391	Default: 141195	Default: 257159	Default: 282391

Personal income (regular/non-AMT/non-pass-through) tax rate 5

0.32

Default: 0.33

Personal income (regular/non-AMT/non-pass-through) tax bracket (upper threshold) 5

single	mjoint	mseparate	headhh	widow
237588.95	475177.9	237588.95	237588.95	475177.9
Default: 504273	Default: 504273	Default: 252137	Default: 504273	Default: 504273

Personal income (regular/non-AMT/non-pass-through) tax rate 6

0.35

Default: 0.35

Personal income (regular/non-AMT/non-pass-through) tax bracket (upper threshold) 6

single	mjoint	mseparate	headhh	widow
593972.38	712766.86	356383.43	593972.38	712766.86
Default: 506331	Default: 569622	Default: 284811	Default: 537976	Default: 569622

Personal income (regular/non-AMT/non-pass-through) tax rate 7

0.37

Default: 0.396

Source: Compute Studio, "PSLmodels/Tax Cruncher," <https://compute.studio/PSLmodels/Tax-Cruncher/291/edit/>.

## Notes

1. Compute Studio, “PSLmodels/Tax-Cruncher,” <https://compute.studio/PSLmodels/Tax-Cruncher/>.
2. GitHub, “How to Use Tax-Cruncher,” <https://github.com/PSLmodels/Tax-Cruncher#how-to-use-tax-cruncher>.
3. GitHub, “Tax-Cruncher,” <https://github.com/PSLmodels/Tax-Cruncher>.
4. Tax-Cruncher relies on Tax-Calculator, a microsimulation model of the US federal individual income and payroll tax system, and ParamTools, a library for parameter processing and validation and other libraries for data processing, analysis, and deployment.
5. Policy Simulation Library, “Models,” <https://www.pslmodels.org/Catalog/index.html>.
6. Notable exceptions include a slower rate of inflation (chained CPI) for parameters of the tax code and the repeal of the health insurance mandate.
7. Refer to Appendix A for a more complete description of the inputs. In addition, inputs to this simulation can be viewed and edited at Compute Studio, <https://compute.studio/PSLmodels/Tax-Cruncher/291/edit/>.
8. Inflation adjustments for future years are estimated using predictions of inflation from the Congressional Budget Office’s 10-year *Budget and Economic Outlook* report. Congress of the United States, Congressional Budget Office, *The Budget and Economic Outlook: 2019 to 2029*, January 2019, <https://www.cbo.gov/system/files/2019-03/54918-Outlook-3.pdf>. We do not apply the IRS’s rounding rules.
9. Refer to Table 3 to see how TCJA repealed the personal exemption.
10. Refer to Table 4 to see how TCJA modified the CTC.
11. The outputs to this simulation can be viewed at Compute Studio, “Results,” <https://compute.studio/PSLmodels/Tax-Cruncher/291/>.
12. Refer to Table 4 to see how TCJA modified the CTC.
13. The personal exemption under 2026 law will be approximately \$4,901 per person, including dependents. Therefore, the personal exemption for our filer totals  $\$4,901 * 4 = \$19,604$ . On the other hand, the standard deduction in 2026 for married filers under the TCJA extension is \$28,510, which is \$8,510 greater than our filer’s deductible expenses of \$20,000. Finally, we can calculate the difference in taxable income by comparing the magnitude of the personal exemption’s effect versus the standard deduction:  $\$19,604 - \$8,510 = \$11,094$ .
14. For the materials for the following case study, see GitHub, “PSLmodels / Tax-Cruncher,” [https://github.com/PSLmodels/Tax-Cruncher/tree/master/docs/WP\\_example](https://github.com/PSLmodels/Tax-Cruncher/tree/master/docs/WP_example).
15. For more information about the input, see GitHub, “PSLmodels / Tax-Cruncher,” [https://github.com/PSLmodels/Tax-Cruncher/blob/master/docs/INPUT\\_INSTRUCTIONS.md](https://github.com/PSLmodels/Tax-Cruncher/blob/master/docs/INPUT_INSTRUCTIONS.md).
16. For instructions specifying a tax reform in a JSON file, see GitHub, “How to Specify a Tax Reform in a JSON Policy Reform File,” <https://github.com/PSLmodels/Tax-Calculator/blob/master/taxcalc/reforms/REFORMS.md#how-to-specify-a-tax-reform-in-a-json-policy-reform-file>.
17. Open Source Policy Center, “Portfolio,” <https://ospc.org/portfolio>.
18. GitHub, “Policy Simulation Library,” <https://github.com/pslmodels>.
19. GitHub, “USA Federal Individual Income and Payroll Tax Microsimulation Model,” <https://github.com/PSLmodels/Tax-Calculator>.
20. GitHub, “A Cost of Capital and Effective Tax Rate Calculator,” <https://github.com/PSLmodels/Cost-of-Capital-Calculator>.
21. While not pictured, income earned from pass-through businesses is taxed at the same rate as personal income and set to identical values in the Compute Studio interface.

© 2020 by the American Enterprise Institute. All rights reserved.

The American Enterprise Institute (AEI) is a nonpartisan, nonprofit, 501(c)(3) educational organization and does not take institutional positions on any issues. The views expressed here are those of the author(s).