



JSON整形・圧縮・チェック

[チートシート](#)

JSONをインデント付きで見やすく整形、1行に圧縮、キーの並べ替え。構文エラーは行と列で教えます。

JSON の型

型	書き方の例	ポイント
文字列	"山田"	ダブルクォートだけ (' は不可)
数値	42 / -1.5 / 1e3	先頭0 (012) ・ NaN ・ Infinity は不可
真偽値	true / false	小文字のみ
null	null	値なし
配列	[1, 2, 3]	順番のある並び
オブジェクト	{"id": 1, "name": "A"}	キーは必ず " " で囲む

文字列の中のエスケープ

書き方	意味
\"	ダブルクォート
\\	バックスラッシュ (Windows のパス C:\\Users)
\n / \r / \t	改行 / 復帰 / タブ
\u3042	Unicode の文字 (あ)

整形・圧縮のコマンド

環境	書き方
JavaScript	JSON.stringify(obj, null, 2)
Python	json.dumps(obj, ensure_ascii=False, indent=2)
コマンド (jq)	jq . data.json / jq -c . data.json
PowerShell	Get-Content a.json ConvertFrom-Json ConvertTo-Json -Depth 10

よくあるエラー

NG	OK	原因
{"a": 1,}	{"a": 1}	最後のカンマ
{'a': 1}	{"a": 1}	シングルクォート
{a: 1}	{"a": 1}	キーに " " がない
{"a": 1} // メモ	{"a": 1}	コメントは書けない
{"a": 01}	{"a": 1}	先頭の0
{"a": undefined}	{"a": null}	undefined は無い
"改行 入り"	"改行\n入り"	文字列の中の改行は \n



SQL整形（フォーマッター）

[チートシート](#)

よく使う SQL の書き方（標準 SQL。DB ごとの違いはカッコ内）。

取り出す（SELECT）

書き方	意味
SELECT * FROM users	すべての列・行
SELECT id, name FROM users	列を指定
WHERE dept = '営業' AND age >= 30	条件
WHERE name LIKE '山%'	前方一致（%は何文字でも）
WHERE id IN (1, 2, 3)	どれかに一致
WHERE age BETWEEN 20 AND 29	範囲（両端を含む）
WHERE memo IS NULL	NULL かどうか（= NULL は不可）
ORDER BY created_at DESC	並べ替え（新しい順）
LIMIT 10 OFFSET 20	件数（SQL Server: TOP / OFFSET FETCH）
SELECT DISTINCT dept FROM users	重複を除く

集計・結合

書き方	意味
SELECT dept, COUNT(*) FROM users GROUP BY dept	グループごとの件数
... GROUP BY dept HAVING COUNT(*) >= 5	集計した結果で絞る
SUM(x) / AVG(x) / MAX(x) / MIN(x)	合計・平均・最大・最小
FROM a INNER JOIN b ON a.id = b.a_id	両方にあるものだけ
FROM a LEFT JOIN b ON a.id = b.a_id	a はすべて、b は合うものだけ
UNION / UNION ALL	結果を縦につなぐ（ALL は重複も残す）
WITH t AS (SELECT ...) SELECT * FROM t	名前付きの副問い合わせ
ROW_NUMBER() OVER (PARTITION BY dept ORDER BY age)	グループ内の連番

追加・更新・削除

書き方	意味
INSERT INTO users (id, name) VALUES (1, '山田')	追加
UPDATE users SET name = '佐藤' WHERE id = 1	更新（WHERE を忘れると全件）
DELETE FROM users WHERE id = 1	削除（WHERE を忘れると全件）
BEGIN; ... COMMIT; / ROLLBACK;	トランザクション

テーブル

書き方	意味
CREATE TABLE t (id INT PRIMARY KEY, name VARCHAR(50))	作る
ALTER TABLE t ADD COLUMN memo TEXT	列を足す
CREATE INDEX idx_t_name ON t (name)	インデックス
DROP TABLE t	消す（元に戻せない）



JSON・YAML・CSV 相互変換

シートシート

YAML の書き方と、同じデータを JSON・CSV で書いたときの対応。

YAML の書き方 → JSON

YAML	JSON
name: 山田 age: 30	{"name": "山田", "age": 30}
tags: - a - b	{"tags": ["a", "b"]}
tags: [a, b]	{"tags": ["a", "b"]}
user: name: 山田 dept: 営業	{"user": {"name": "山田", "dept": "営業"}}
- id: 1 - id: 2	[{"id": 1}, {"id": 2}]
ok: true memo: null	{"ok": true, "memo": null}
text: 1行目 2行目	{"text": "1行目\n2行目\n"}
# コメント	(JSON には無い)

YAML の注意

書き方	結果	対策
zip: 0123	数値や8進数になることがある	zip: "0123"
answer: yes	真偽値になることがある (YAML 1.1)	answer: "yes"
time: 1:30	数値になることがある (YAML 1.1)	time: "1:30"
インデントにタブ	エラー	スペース2つ
key:value	1つの文字列	コロンの後ろに空白

JSON ⇔ CSV の対応

JSON	CSV
[{"id":1,"name":"A"}]	id,name 1,A
[{"user":{"name":"A"}}]	user.name A
[{"tags":["a","b"]}]	tags ["a","b"]



ファイルの作成日時・更新日時の変更

シートシート

ファイルの作成日時・更新日時を好きな日時に。PowerShellのコマンド作成、日時を指定したZIP、PDF・Office・写真の中の日時の書き換えに対応。

PowerShell で日時を変える

やりたいこと	書き方
作成日時	<code>(Get-Item 'a.xlsx').CreationTime = '2025-04-01 09:00'</code>
更新日時	<code>(Get-Item 'a.xlsx').LastWriteTime = '2025-04-01 09:00'</code>
アクセス日時	<code>(Get-Item 'a.xlsx').LastAccessTime = '2025-04-01 09:00'</code>
フォルダ内すべて	<code>Get-ChildItem -File % { \$_.LastWriteTime = '2025-04-01 09:00' }</code>
今の日時を確認	<code>Get-Item 'a.xlsx' Select Name, CreationTime, LastWriteTime</code>

Mac / Linux

やりたいこと	書き方
更新日時	<code>touch -m -t 202504010900 a.xlsx</code>
今の日時に	<code>touch a.xlsx</code>
作成日時 (Mac)	<code>SetFile -d '04/01/2025 09:00:00' a.xlsx</code>
確認	<code>stat a.xlsx</code>

ファイルの中の日時

形式	場所
Word・Excel	ファイル → 情報 → プロパティ
PDF	文書のプロパティ (作成日・更新日)
写真 (JPEG)	Exif の撮影日時



全角⇄半角変換

シートシート

英数字・記号・スペース・カタカナを、種類を選んで全角⇄半角に変換。半角カナの濁点もきちんと処理します。

全角と半角の対応

種類	全角	半角
英字	A B C a b c	ABC abc
数字	0 1 2 3	0123
記号	! # \$ % & ()	!#\$%&()
スペース	(全角スペース)	(半角スペース)
カタカナ	アイウ	ｱｲｳ
濁点	ガ (1文字)	が (2文字)
長音	ー	-
円記号	¥	＼ (日本語環境では ¥ に見える)

よくある使い分け

データ	そろえ方
電話番号・郵便番号	半角数字＋ハイフン
住所の番地	半角数字 (1-2-3)
氏名のフリガナ	全角カタカナ
銀行の振込名義	半角カナ (大文字のみ)
メールアドレス・URL	半角英数字
商品名・文章	英数字は半角、カタカナは全角

Excel の関数

書き方	意味
=ASC(A1)	全角 → 半角 (カナも半角に)
=JIS(A1)	半角 → 全角
=PHONETIC(A1)	ふりがなを取り出す
=TRIM(A1)	前後の空白を消し、間の空白を1つに
=SUBSTITUTE(A1," "," ")	全角スペースを半角に



環境依存文字（機種依存文字）チェッカー

シートシート

①・(株)などは、Mac や別のシステム・メールで化けることがあります。置き換えの目安です。

丸数字・ローマ数字

文字	置き換え	文字	置き換え
①②③	(1)(2)(3)	I II III	I II III
⑩	(10)	i ii iii	i ii iii

組文字・単位

文字	置き換え	文字	置き換え
(株)	(株)	(有)	(有)
(代)	(代)	TEL	TEL
No.	No.	平成	平成
昭和	昭和	令和	令和
cm	cm	kg	kg
m ²	m2 / 平方メートル	ミリ	ミリ
キロ	キロ	センチ	センチ

そのほか注意する文字

文字	問題	対策
半角カナ アイ	メール・Web で化けること がある	全角カナに
高・崎・徳（異体 字）	Shift_JIS に無 い（CP932 だ け）	高・崎・徳 に（名前 は本人に確認）
（つちよし）	Shift_JIS に無 い	吉 に（名前は本人に 確認）
絵文字 🍌	古いシステム で？に化ける	使わない
～（波ダッシュ）	Windows と Mac で別の文 字に	～（全角チルダ）に そろえる



和暦⇄西暦変換・年齢早見表

2026年版。年齢はその年の誕生日を迎えたあとの満年齢です。

シートシート

元号と西暦の換算

元号	期間	西暦 → 和暦	和暦 → 西暦
令和	2019/5/1～	西暦 - 2018	令和 + 2018
平成	1989/1/8～ 2019/4/30	西暦 - 1988	平成 + 1988
昭和	1926/12/25 ～1989/1/7	西暦 - 1925	昭和 + 1925
大正	1912/7/30 ～ 1926/12/24	西暦 - 1911	大正 + 1911
明治	1868/1/25 ～ 1912/7/29	西暦 - 1867	明治 + 1867

西暦・和暦・年齢（2026年）（1/5）

西暦	和暦	干支	年齢
2026	[object Object]8年	丙午	0歳
2025	[object Object]7年	乙巳	1歳
2024	[object Object]6年	甲辰	2歳
2023	[object Object]5年	癸卯	3歳
2022	[object Object]4年	壬寅	4歳
2021	[object Object]3年	辛丑	5歳
2020	[object Object]2年	庚子	6歳
2019	[object Object]元年	己亥	7歳
2018	[object Object]30年	戊戌	8歳
2017	[object Object]29年	丁酉	9歳
2016	[object Object]28年	丙申	10歳
2015	[object Object]27年	乙未	11歳
2014	[object Object]26年	甲午	12歳
2013	[object Object]25年	癸巳	13歳
2012	[object Object]24年	壬辰	14歳
2011	[object Object]23年	辛卯	15歳
2010	[object Object]22年	庚寅	16歳
2009	[object Object]21年	己丑	17歳
2008	[object Object]20年	戊子	18歳
2007	[object Object]19年	丁亥	19歳

西暦・和暦・年齢（2026年）（2/5）

西暦	和暦	干支	年齢
2006	[object Object]18年	丙戌	20歳
2005	[object Object]17年	乙酉	21歳
2004	[object Object]16年	甲申	22歳
2003	[object Object]15年	癸未	23歳
2002	[object Object]14年	壬午	24歳
2001	[object Object]13年	辛巳	25歳
2000	[object Object]12年	庚辰	26歳
1999	[object Object]11年	己卯	27歳
1998	[object Object]10年	戊寅	28歳
1997	[object Object]9年	丁丑	29歳
1996	[object Object]8年	丙子	30歳
1995	[object Object]7年	乙亥	31歳
1994	[object Object]6年	甲戌	32歳
1993	[object Object]5年	癸酉	33歳
1992	[object Object]4年	壬申	34歳
1991	[object Object]3年	辛未	35歳
1990	[object Object]2年	庚午	36歳
1989	[object Object]元年	己巳	37歳
1988	[object Object]63年	戊辰	38歳
1987	[object Object]62年	丁卯	39歳

西暦・和暦・年齢（2026年）（3/5）

西暦	和暦	干支	年齢
1986	[object Object]61年	丙寅	40歳
1985	[object Object]60年	乙丑	41歳
1984	[object Object]59年	甲子	42歳
1983	[object Object]58年	癸亥	43歳
1982	[object Object]57年	壬戌	44歳
1981	[object Object]56年	辛酉	45歳
1980	[object Object]55年	庚申	46歳
1979	[object Object]54年	己未	47歳
1978	[object Object]53年	戊午	48歳
1977	[object Object]52年	丁巳	49歳
1976	[object Object]51年	丙辰	50歳
1975	[object Object]50年	乙卯	51歳
1974	[object Object]49年	甲寅	52歳
1973	[object Object]48年	癸丑	53歳
1972	[object Object]47年	壬子	54歳
1971	[object Object]46年	辛亥	55歳
1970	[object Object]45年	庚戌	56歳
1969	[object Object]44年	己酉	57歳
1968	[object Object]43年	戊申	58歳
1967	[object Object]42年	丁未	59歳

西暦・和暦・年齢（2026年）（4/5）

西暦	和暦	干支	年齢
1966	[object Object]41年	丙午	60歳
1965	[object Object]40年	乙巳	61歳
1964	[object Object]39年	甲辰	62歳
1963	[object Object]38年	癸卯	63歳
1962	[object Object]37年	壬寅	64歳
1961	[object Object]36年	辛丑	65歳
1960	[object Object]35年	庚子	66歳
1959	[object Object]34年	己亥	67歳
1958	[object Object]33年	戊戌	68歳
1957	[object Object]32年	丁酉	69歳
1956	[object Object]31年	丙申	70歳
1955	[object Object]30年	乙未	71歳
1954	[object Object]29年	甲午	72歳
1953	[object Object]28年	癸巳	73歳
1952	[object Object]27年	壬辰	74歳
1951	[object Object]26年	辛卯	75歳
1950	[object Object]25年	庚寅	76歳
1949	[object Object]24年	己丑	77歳
1948	[object Object]23年	戊子	78歳
1947	[object Object]22年	丁亥	79歳

西暦・和暦・年齢（2026年）（5/5）

西暦	和暦	干支	年齢
1946	[object Object]21年	丙戌	80歳
1945	[object Object]20年	乙酉	81歳
1944	[object Object]19年	甲申	82歳
1943	[object Object]18年	癸未	83歳
1942	[object Object]17年	壬午	84歳
1941	[object Object]16年	辛巳	85歳
1940	[object Object]15年	庚辰	86歳
1939	[object Object]14年	己卯	87歳
1938	[object Object]13年	戊寅	88歳
1937	[object Object]12年	丁丑	89歳
1936	[object Object]11年	丙子	90歳
1935	[object Object]10年	乙亥	91歳
1934	[object Object]9年	甲戌	92歳
1933	[object Object]8年	癸酉	93歳
1932	[object Object]7年	壬申	94歳
1931	[object Object]6年	辛未	95歳
1930	[object Object]5年	庚午	96歳
1929	[object Object]4年	己巳	97歳
1928	[object Object]3年	戊辰	98歳
1927	[object Object]2年	丁卯	99歳



営業日計算（日本の祝日対応）

シートシート

2026年・2027年の祝日と、Excel で営業日を数える関数。

2026年の祝日

日付	曜日	祝日
1/1	木	元日
1/12	月	成人の日
2/11	水	建国記念の日
2/23	月	天皇誕生日
3/20	金	春分の日
4/29	水	昭和の日
5/3	日	憲法記念日
5/4	月	みどりの日
5/5	火	こどもの日
5/6	水	振替休日
7/20	月	海の日
8/11	火	山の日
9/21	月	敬老の日
9/22	火	国民の休日
9/23	水	秋分の日
10/12	月	スポーツの日
11/3	火	文化の日
11/23	月	勤労感謝の日

2027年の祝日

日付	曜日	祝日
1/1	金	元日
1/11	月	成人の日
2/11	木	建国記念の日
2/23	火	天皇誕生日
3/21	日	春分の日
3/22	月	振替休日
4/29	木	昭和の日
5/3	月	憲法記念日
5/4	火	みどりの日
5/5	水	こどもの日
7/19	月	海の日
8/11	水	山の日
9/20	月	敬老の日
9/23	木	秋分の日
10/11	月	スポーツの日
11/3	水	文化の日
11/23	火	勤労感謝の日

Excel の営業日の関数

関数	意味
=NETWORKDAYS(開始, 終了, 祝日)	期間の営業日数（土日・祝日を除く）
=WORKDAY(開始, 日数, 祝日)	○営業日後の日付
=WORKDAY(開始, -日数, 祝日)	○営業日前の日付
=NETWORKDAYS.INTL(開始, 終了, "0000011")	休みの曜日を指定（1が休み。月～日の順）
=WORKDAY(EOMONTH(A1, 0)+1, -1, 祝日)	月末の最終営業日

💡 「祝日」には祝日の日付を並べた範囲（例: \$H\$2:\$H\$30）を指定



日付・日数計算（○日後・期間の計算）

シートシート

2つの日付の日数や「○年○か月○日」、○日後・○か月後の日付を計算。曜日・和暦・祝日も表示。

Excel の日付の関数

書き方	意味	例（A1 = 2025/1/31）
=TODAY()	今日の日付	—
=NOW()	今の日時	—
=A1+30	30日後	2025/3/2
=EDATE(A1,1)	1か月後（月末は末日）	2025/2/28
=EOMONTH(A1,0)	その月の末日	2025/1/31
=EOMONTH(A1,-1)+1	その月の1日	2025/1/1
=DATE(2025,4,1)	年・月・日から日付	2025/4/1
=B1-A1	2つの日付の日数	—
=DATEDIF(A1,B1,"Y")	満年数（年齢・勤続年数）	—
=DATEDIF(A1,B1,"YM")	1年未満の月数	—
=WEEKDAY(A1)	曜日の番号（日=1～土=7）	6
=TEXT(A1,"aaa")	曜日（金）	金
=TEXT(A1,"ggge年")	和暦（令和7年）	令和7年
=YEAR(A1) / MONTH / DAY	年・月・日を取り出す	2025 / 1 / 31

表示形式（ユーザー定義）

書式	表示
yyyy/mm/dd	2025/01/31
yyyy"年"m"月"d"日"	2025年1月31日
m/d(aaa)	1/31(金)
ggge"年"m"月"d"日"	令和7年1月31日
[h]:mm	25:30（24時間を超える合計）

日付の数え方

言い方	意味
初日不算入（民法）	翌日から数える。4/1 から1か月 → 5/1 まで
「○日以内」	起算日から数えて○日目まで
「○日前まで」	その日を含めず、○日さかのぼった日まで
月末の○か月後	1/31 の1か月後 → 2/28（末日に丸める）



漢数字変換（大字・領収書の金額表記）

シートシート

数字⇄漢数字の変換。壹・弍・参の大字や「金壹万円也」のような領収書・契約書のアmount表記にも対応。

数字・漢数字・大字

数字	漢数字	大字（領収書）
1	一	壹
2	二	弍
3	三	参
4	四	四（肆）
5	五	五（伍）
6	六	六（陸）
7	七	七（漆）
8	八	八（捌）
9	九	九（玖）
10	十	拾
100	百	百（佰）
1,000	千	千（阡・仟）
10,000	万	萬

💡 一般には 壹・弍・参・拾・萬 だけを大字にします（四～九・百・千は普通の漢字でよい）

金額の書き方

金額	書き方
10,000円	金壹万円也
30,000円	金参万円也
50,000円	金五万円也
108,000円	金壹拾万八千円也
1,234,567円	金壹百弍拾参万四千五百六拾七円也

💡 頭に「金」、終わりに「也」（または「円」の後ろに「也」）を付けて、書き足しを防ぎます

大きな数の位

位	数
万	10^4
億	10^8
兆	10^{12}
京	10^{16}
割・分・厘	$0.1 \cdot 0.01 \cdot 0.001$



振込名義・全銀カナ変換（半角カナ・法人略語）

シートシート

「株式会社ヤマダ」→「カヤマダ」のように、銀行振込で使える半角カナに変換。小さい文字・長音・使えない記号もチェックします。

法人格の略語

法人格	略語
特定非営利活動法人	トクニ
一般社団法人	シャ
公益社団法人	シャ
一般財団法人	ザイ
公益財団法人	ザイ
社会福祉法人	フク
司法書士法人	シホリ
行政書士法人	ギョウ
弁護士法人	ヘン
税理士法人	ゼイ
株式会社	カ
有限会社	ユ
合同会社	ト
合名会社	メ
合資会社	シ
医療法人	イ
学校法人	ガク
宗教法人	シユウ
監査法人	カンサ

カッコの付け方

位置	書き方	例
前（前株）	略語)	カヤマダ [®] ショウ [®]
後ろ（後株）	(略語	ヤマダ [®] ショウジ [®] (カ
途中	(略語)	ヤマダ [®] (カ)トリキョウシテン

使える文字・置き換え

入力	振込名義
ヤユヨツ（小さい文字）	ヤユヨツ（大きく）
アイウエオ	アイウエオ
ー（長音）	-（ハイフン）
・（中黒）	.（ピリオド）
ヴ	ヴ
ヲ	ヲ
a b c（英字）	ABC（半角大文字）
使える記号	() . - / , 「 」 \ と空白
漢字・ひらがな	使えない（カナで）



電話番号・郵便番号のハイフン整形

シートシート

「0312345678」→「03-1234-5678」のように、市外局番の長さに合わせてハイフンを入れます。外す・全角・+81形式にも。

番号の種類と区切り方

種類	先頭	区切り方
携帯電話	090・080・070	090-1234-5678
IP電話	050	050-1234-5678
固定（東京・大阪）	03・06	03-1234-5678
固定（市外局番3桁）	011・045・052 など	045-123-4567
固定（市外局番4桁）	0466・0995 など	0466-12-3456
固定（市外局番5桁）	01267 など	01267-1-2345
フリーダイヤル	0120	0120-123-456
フリーダイヤル	0800	0800-123-4567
ナビダイヤル	0570	0570-012-345
郵便番号	7桁	100-0001

国際表記（+81）

国内	国際
03-1234-5678	+81-3-1234-5678
090-1234-5678	+81-90-1234-5678
0120-123-456	（海外からはつながらないことが多い）
💡 先頭の 0 を取って +81 を付けます（+81-03 は間違い）	



勤務時間・残業時間の計算（深夜・15分単位の丸め）

シートシート

出勤・退勤・休憩から実働時間を計算。残業・深夜（22～5時）も分けて出し、1か月分の合計や10進数の時間（7.5h）も。

割増賃金（最低）

区分	どの時間	割増
時間外	1日8時間・週40時間を超えた分	25%
深夜	22時～翌5時	25%
時間外＋深夜	残業が深夜にかかった分	50%
法定休日	週1日の休日に働いた分	35%
法定休日＋深夜		60%
月60時間超の時間外		50%

分と10進数

分	時間	分	時間
5	0.08	35	0.58
10	0.17	40	0.67
15	0.25	45	0.75
20	0.33	50	0.83
25	0.42	55	0.92
30	0.5	60	1.0

休憩（労働基準法）

労働時間	休憩
6時間まで	なくてもよい
6時間を超える	45分以上
8時間を超える	1時間以上

Excel の時間計算

書き方	意味
=B1-A1-C1	退勤 － 出勤 － 休憩
=(B1-A1-C1)*24	10進数の時間（表示形式は標準）
=MAX(0,D1-"8:00")	8時間を超えた残業
=FLOOR(A1,"0:15")	15分単位で切り捨て
=CEILING(A1,"0:15")	15分単位で切り上げ
[h]:mm	24時間を超える合計の表示形式



URLエンコード・デコード

[チートシート](#)

日本語URLのパーセントエンコードと復元。文字化けしたURLを読める形に戻せます。

よくエンコードされる文字

文字	エンコード後	文字	エンコード後
(空白)	%20 / +	!	%21
"	%22	#	%23
\$	%24	%	%25
&	%26	'	%27
+	%2B	,	%2C
/	%2F	:	%3A
;	%3B	=	%3D
?	%3F	@	%40
あ	%E3%81%82	日本	%E6%97%A5%E6%9C%AC

エンコードしなくてよい文字

種類	文字
英数字	A-Z a-z 0-9
記号 (RFC 3986)	- _ . ~

言語ごとの関数

言語	エンコード	デコード
JavaScript (値)	<code>encodeURIComponent(s)</code>	<code>decodeURIComponent(s)</code>
JavaScript (URL全体)	<code>encodeURIComponent(url)</code>	<code>decodeURIComponent(url)</code>
Python	<code>urllib.parse.quote(s)</code>	<code>urllib.parse.unquote(s)</code>
PHP	<code>rawurlencode(\$s)</code>	<code>rawurldecode(\$s)</code>
Java	<code>URLEncoder.encode(s, UTF_8)</code>	<code>URLDecoder.decode(s, UTF_8)</code>
Excel	<code>=ENCODEURL(A1)</code>	(関数なし)
PowerShell	<code>[uri]::EscapeDataString(\$s)</code>	<code>[uri]::UnescapeDataString(\$s)</code>



UNIX時間⇄日時変換（日本時間）

シートシート

UNIXタイムスタンプ（秒・ミリ秒）を日本時間の日時に、日時をUNIX時間に変換。ISO 8601表記も出します。

今のUNIX時間を取る

環境	書き方
JavaScript (秒)	<code>Math.floor(Date.now() / 1000)</code>
Python	<code>int(time.time())</code>
PHP	<code>time()</code>
Java	<code>Instant.now().getEpochSecond()</code>
Go	<code>time.Now().Unix()</code>
Linux / Mac	<code>date +%s</code>
PowerShell	<code>[DateTimeOffset]::Now.ToUnixTimeSeconds()</code>
MySQL	<code>UNIX_TIMESTAMP()</code>
PostgreSQL	<code>EXTRACT(EPOCH FROM now())</code>

UNIX時間 → 日時

環境	書き方
JavaScript	<code>new Date(sec * 1000)</code>
Python	<code>datetime.fromtimestamp(sec, ZoneInfo("Asia/Tokyo"))</code>
Linux	<code>date -d @1700000000</code>
Mac	<code>date -r 1700000000</code>
MySQL	<code>FROM_UNIXTIME(sec)</code>
Excel（日本時間）	<code>=A1/86400+DATE(1970,1,1)+9/24</code>

目安になる値

UNIX時間	日本時間
0	1970/01/01 09:00:00
1000000000	2001/09/09 10:46:40
1700000000	2023/11/15 07:13:20
2000000000	2033/05/18 12:33:20
2147483647	2038/01/19 12:14:07（32bit の上限）
💡 秒は10桁、ミリ秒は13桁。1日 = 86,400秒	



正規表現テスター

シートシート

JavaScript の正規表現（多くの言語・エディタではほぼ共通）。例の「マッチ」は、その書き方で見つかる文字です。

文字の種類

書き方	意味	例 → マッチ
.	改行以外の1文字	a.c → abc, a-c
\d	数字（0～9）	\d\d → 42
\D	数字以外	\D → a, あ
\w	英数字と _	\w+ → user_01
\s	空白・タブ・改行	a\s b → a b
[abc]	a・b・c のどれか	[aeiou] → e
[^abc]	a・b・c 以外	[^0-9] → x
[a-z]	a～z の範囲	[A-Z]+ → ABC
\.	記号そのもの（エスケープ）	\. → .

回数

書き方	意味	例 → マッチ
*	0回以上	ab* → a, abbb
+	1回以上	ab+ → ab, abbb
?	0回か1回	colou?r → color
{3}	ちょうど3回	\d{3} → 123
{2,4}	2～4回	\d{2,4} → 2025
{2,}	2回以上	a{2,} → aaa
*? / +?	できるだけ短く（最短一致）	<.+?> →

位置・グループ

書き方	意味	例 → マッチ
^	行の先頭	^abc → 先頭の abc
\$	行の末尾	\.csv\$ → .csv で終わる
\b	単語の区切り	\bcat\b → cat (category はx)
(abc)	グループ（置換で \$1）	(\d+)円 → \$1 で数字だけ
(?:abc)	番号を付けないグループ	(?:ab)+ → abab
a b	a または b	猫 犬 → 犬
(?=...)	後ろに...が続く	\d+(?=円) → 100円の100
(?!...)	後ろに...が続かない	a(?!b) → ac の a
(?<=...)	前に...がある	(?<=¥)\d+ → ¥500の500

日本語でよく使うパターン

対象	正規表現
ひらがな	[あ-んー]+
カタカナ	[ア-ケー]+
半角カナ	[ㇿ°]+
漢字	[一-龠々]+
全角英数字	[A-Z a-z 0-9]+
郵便番号	\d{3}-?\d{4}
携帯電話番号	0[789]0-?\d{4}-?\d{4}
メールアドレス（簡易）	[\w.+-]+@[\w-]+(\. [\w-]+)+
日付 2025/04/01	\d{4}[/-]\d{1,2}[/-]\d{1,2}
全角・半角スペース	[\u3000]+

フラグ

フラグ	意味
g	すべて探す（1つ目で止めない）
i	大文字・小文字を区別しない
m	^ と \$ を各行の先頭・末尾にする
s	. が改行にもマッチする
u	Unicode として扱う（絵文字など4バイトの文字）



進数変換（2進数・8進数・10進数・16進数）

シートシート

2進数・8進数・10進数・16進数を相互に変換。大きな数や負の数（2の補数）、2〜36進数にも対応。

0〜16 の対応

10進	2進	8進	16進
0	0000	0	0
1	0001	1	1
2	0010	2	2
3	0011	3	3
4	0100	4	4
5	0101	5	5
6	0110	6	6
7	0111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F
16	10000	20	10

よく出てくる数

10進	16進	意味
255	FF	1バイトの最大
256	100	1バイトで表せる数
1,024	400	1KiB
65,535	FFFF	2バイトの最大
2,147,483,647	7FFFFFFF	32bit 整数の最大
4,294,967,295	FFFFFFFF	32bit 符号なしの最大

プログラムでの書き方

書き方	意味	変換
0b1010	2進数 (10)	n.toString(2)
0o12	8進数 (10)	n.toString(8)
0xA	16進数 (10)	n.toString(16)
parseInt("ff", 16)	16進の文字列 → 数	255
=DEC2HEX(255) / =HEX2DEC("FF")	Excel	FF / 255



カラーコード変換（HEX・RGB・HSL）

[チートシート](#)

HEX・RGB・HSL・HSVを相互変換。文字と背景のコントラスト比の判定や、濃淡のバリエーションも。

色の書き方（CSS）

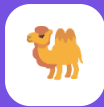
形式	書き方	ポイント
HEX	#2563c9	赤緑青を16進2桁ずつ
HEX（短縮）	#fff	#ffffffと同じ
HEX（透明度）	#2563c980	最後の2桁が透明度
RGB	rgb(37 99 201)	0～255
RGB（透明度）	rgb(37 99 201 / 50%)	ID: rgba(37,99,201,.5)
HSL	hsl(217 69% 47%)	色相・彩度・明るさ
名前	tomato / navy	147色の名前

よく使う色

名前	HEX	RGB
白	#ffffff	255 255 255
黒	#000000	0 0 0
グレー	#808080	128 128 128
赤	#ff0000	255 0 0
緑	#008000	0 128 0
青	#0000ff	0 0 255
黄	#ffff00	255 255 0
オレンジ	#ffa500	255 165 0

文字と背景のコントラスト比（WCAG）

基準	普通の文字	大きい文字
AA（最低限）	4.5 : 1 以上	3 : 1 以上
AAA（より読みやすい）	7 : 1 以上	4.5 : 1 以上
💡 大きい文字＝18pt 以上、または太字で 14pt 以上		



キャメルケース・スネークケース変換

チートシート

camelCase・snake_case・kebab-caseなど、変数名や列名の命名規則をまとめて変換します。

命名規則の種類

名前	例	よく使う場所
camelCase (キャメル)	userName	JavaScript・Java の変数・関数
PascalCase (パスカル)	UserName	クラス名・型名・C# のメソッド
snake_case (スネーク)	user_name	Python・Ruby・DB の列名
SCREAMING_SNAKE	USER_NAME	定数・環境変数
kebab-case (ケバブ)	user-name	URL・CSS のクラス名・ファイル名
dot.case	user.name	設定のキー
Train-Case	User-Name	HTTP ヘッダー
lowercase	username	パッケージ名

言語ごとの慣習

言語	変数・関数	クラス	定数
JavaScript / TypeScript	camelCase	PascalCase	UPPER_SNAKE
Java	camelCase	PascalCase	UPPER_SNAKE
Python	snake_case	PascalCase	UPPER_SNAKE
C#	camelCase (ローカル)	PascalCase	PascalCase
Go	camelCase (公開は Pascal)	PascalCase	PascalCase
Ruby	snake_case	PascalCase	UPPER_SNAKE
SQL	snake_case	—	—
CSS	kebab-case	—	—



データ容量の単位変換・転送時間の計算

[チートシート](#)

KB・MB・GB・TBとKiB・MiB・GiBを相互変換。回線速度からダウンロードにかかる時間も計算します。

単位

単位	バイト数	単位	バイト数
1 KB	1,000	1 KiB	1,024
1 MB	1,000,000	1 MiB	1,048,576
1 GB	10億	1 GiB	約10.7億
1 TB	1兆	1 TiB	約1.1兆

💡 Windows は「GB」と表示しても中身は GiB（1,024 単位）。だから 1TB の HDD は約 931GB と表示されます

ファイルの大きさの目安

もの	大きさ
メール1通（文字だけ）	数KB
スマホの写真1枚	2～5MB
Excel・Word	数十KB～数MB
PDF（10ページ）	数百KB～数MB
動画1分（フルHD）	約100MB
メールの添付の上限	多くは 10～25MB

ダウンロードにかかる時間（目安）

回線	1GB	10GB
10 Mbps	約14分	約2時間20分
100 Mbps	約1分20秒	約14分
1 Gbps	約8秒	約1分20秒

💡 bps はビット。1バイト = 8ビット（100Mbps ≒ 12.5MB/秒）



日本時間変換（時差計算・世界時計）

チートシート

2026年版。マイナスは日本より遅れている時間（日本が 22:00 なら -13 の都市は 9:00）。

主な都市と日本との時差

都市	冬	夏（夏時間）
ロサンゼルス	-17	-16
デンバー	-16	-15
シカゴ	-15	-14
ニューヨーク	-14	-13
サンパウロ	-12	-12
ロンドン	-9	-8
パリ・ベルリン	-8	-7
モスクワ	-6	-6
ドバイ	-5	-5
インド	-3.5	-3.5
バンコク・ジャカルタ	-2	-2
シンガポール・上海・台北	-1	-1
ソウル	±0	±0
シドニー	+2（夏時間）	+1
オークランド	+4（夏時間）	+3

💡 「夏」は北半球の夏（7月）。南半球のシドニーなどは日本の冬が夏時間

タイムゾーンの略号

略号	意味	UTC との差
JST	日本標準時	+9
UTC / GMT	協定世界時 / グリニッジ標準時	0
PST / PDT	太平洋時間（標準 / 夏）	-8 / -7
EST / EDT	東部時間（標準 / 夏）	-5 / -4
CET / CEST	中央ヨーロッパ時間	+1 / +2
BST	イギリス夏時間	+1
IST	インド標準時	+5:30

サマータイムの期間

地域	始まり	終わり
アメリカ・カナダ	3月第2日曜	11月第1日曜
ヨーロッパ	3月最終日曜	10月最終日曜
オーストラリア南東部	10月第1日曜	4月第1日曜



cron式の日本語説明・次回実行日時

チートシート

crontab・GitHub Actions・Kubernetes の CronJob などでも共通の「分 時 日 月 曜日」の書き方。

5つの項目

位置	項目	使える値
1	分	0～59
2	時	0～23
3	日	1～31
4	月	1～12 (JAN～DEC)
5	曜日	0～7 (0と7は日曜。SUN～SAT)

記号

記号	意味	例 → 意味
*	すべて	* * * * * → 毎分
,	並べる	0,30 * * * * * → 毎時0分と30分
-	範囲	0 9 * * * 1-5 → 平日9:00
/	間隔	* / 15 * * * * * → 15分ごと

よく使う書き方

cron	意味
* * * * *	毎分
* / 5 * * * *	5分ごと
0 * * * *	毎時0分
0 * / 2 * * *	2時間ごと (0:00, 2:00 ...)
0 9 * * *	毎日 9:00
30 2 * * *	毎日 2:30 (夜間バッチ)
0 9 * * 1-5	平日 9:00
0 9-18 * * 1-5	平日 9～18時の毎時0分
0 18 * * 5	毎週金曜 18:00
0 0 * * 0	毎週日曜 0:00
0 0 1 * *	毎月1日 0:00
0 0 1,15 * *	毎月1日と15日 0:00
0 0 1 1,4,7,10 *	四半期の初日 0:00
0 0 1 1 *	毎年1月1日 0:00

別名 (@ で始まる書き方)

書き方	同じ意味の cron
@hourly	0 * * * *
@daily	0 0 * * *
@weekly	0 0 * * 0
@monthly	0 0 1 * *
@yearly	0 0 1 1 *
@reboot	(起動したときに1回)

注意

ポイント	内容
日と曜日を両方指定	どちらかに当てはまれば実行 (OR)。 0 0 1 * 1 は「毎月1日」と「毎週月曜」
タイムゾーン	サーバーが UTC なら日本時間より9時間遅れ。JST 9:00 は UTC で 0 0 * * *
月末	5項目では書けない。28-31日に動かしてスクリプトで「明日が1日か」を判定
GitHub Actions	UTC で動き、混雑時は数分遅れることがある



JWTデコーダー（中身と有効期限の確認）

チートシート

JWT（JSON Web Token）のヘッダーとペイロードを表示。exp・iatを日本時間にして、期限切れかどうかもわかります。トークンは送信しません。

JWTの形

部分	中身	例
ヘッダー	アルゴリズムと種類	{"alg":"HS256","typ":"JWT"}
ペイロード	ユーザーや期限などのクレーム	{"sub":"123","exp":1735657200}
署名	改ざんされていないことの証明	HMAC / RSA / ECDSA

💡 3つをBase64URLにしてドットでつなぐ：xxxxx.yyyyy.zzzzz（中身は暗号化されていない）

アルゴリズム（alg）

alg	方式	検証に使う鍵
HS256	HMAC + SHA-256	共通の秘密鍵
RS256	RSA 署名	公開鍵（JWKS）
ES256	楕円曲線 ECDSA	公開鍵
none	署名なし	受け付けてはいけない

よくあるトラブル

症状	確認すること
401 になる	exp が切れていないか、aud・iss が合っているか
すぐログアウトする	exp が短すぎないか、時計のずれ
署名エラー	鍵・alg が発行側と同じか

登録済みクレーム

クレーム	意味	値の例
iss	発行者	"https://auth.example.com"
sub	対象（ユーザーID）	"user_123"
aud	受け取り手	"my-api"
exp	有効期限（UNIX時間）	1735657200
nbf	この時刻より前は無効	1735653600
iat	発行した時刻	1735653600
jti	トークンのID	"a1b2c3"



サブネット計算 (IPアドレス・CIDR)

チートシート

「192.168.1.10/24」からネットワークアドレス・ブロードキャスト・使えるIPの範囲・台数を計算。サブネットマスクとの変換や分割も。

サブネットマスク早見表

CIDR	サブネットマスク	アドレス数	使えるホスト数
/8	255.0.0.0	16,777,216	16,777,214
/12	255.240.0.0	1,048,576	1,048,574
/16	255.255.0.0	65,536	65,534
/20	255.255.240.0	4,096	4,094
/21	255.255.248.0	2,048	2,046
/22	255.255.252.0	1,024	1,022
/23	255.255.254.0	512	510
/24	255.255.255.0	256	254
/25	255.255.255.128	128	126
/26	255.255.255.192	64	62
/27	255.255.255.224	32	30
/28	255.255.255.240	16	14
/29	255.255.255.248	8	6
/30	255.255.255.252	4	2
/31	255.255.255.254	2	2
/32	255.255.255.255	1	1

プライベートIPアドレス

CIDR	範囲	よく使う場所
10.0.0.0/8	10.0.0.0～ 10.255.255.255	企業・クラウドの VPC
172.16.0.0/12	172.16.0.0～ 172.31.255.255	企業・Docker
192.168.0.0/16	192.168.0.0～ 192.168.255.255	家庭・小さなオフィ ス

特別なアドレス

アドレス	意味
127.0.0.1	自分自身 (localhost)
0.0.0.0	すべてのアドレス (待ち受け)
169.254.x.x	DHCP で取れなかったとき
100.64.0.0/10	プロバイダーの共有アドレス (CGNAT)
224.0.0.0/4	マルチキャスト
255.255.255.255	ブロードキャスト (全員宛て)

計算の考え方 (192.168.1.10/24 の例)

項目	求め方	例
ネットワークアドレス	IP と マスク の AND	192.168.1.0
ブロードキャスト	ネットワーク の ホスト 部をすべて1	192.168.1.255
使える範囲	ネットワーク+1 ～ ブロードキャスト-1	192.168.1.1～254
ホスト数	$2^{(32-\text{プレフィックス})} - 2$	254



Excelの列番号⇄アルファベット変換（A1・R1C1）

シートシート

28 ⇄ AB のように、Excelの列番号と列名を変換。A1形式とR1C1形式のセル番地もまとめて変換できます。

列番号と列名

番号	列名	番号	列名
1	A	27	AA
10	J	52	AZ
26	Z	53	BA
100	CV	256	IV (旧 .xls の最後)
702	ZZ	703	AAA
1,000	ALL	16,384	XFD (最後の列)

変換の関数

やりたいこと	書き方
番号 → 列名	=SUBSTITUTE(ADDRESS(1,A1,4),"1","")
列名 → 番号	=COLUMN(INDIRECT(A18"1"))
今の列の番号	=COLUMN()
VBA で列名	Split(Cells(1, n).Address(True, False), "\$")(0)
Python (openpyxl)	get_column_letter(n) / column_index_from_string("AB")

セル番地の書き方

書き方	意味
A1	相対参照（コピーするとずれる）
\$A\$1	絶対参照（F4 キーで切り替え）
\$A1 / A\$1	列だけ・行だけ固定
R1C1	R1C1 形式（行1・列1）
R[1]C[-1]	1行下・1列左（相対）
Sheet2!A1	別のシート
'売上 2025'!A1	シート名に空白があるとき



HTMLエスケープ・アンエスケープ

チートシート

< > & " を < > & " に変換、またはその逆。ブログにコードを載せるときや、& だらけの文字を戻すときに。

必ずエスケープする文字

文字	文字参照	数値参照	理由
&	&	&	文字参照の始まり（いちばん最初に変換）
<	<	<	タグの始まり
>	>	>	タグの終わり
"	"	"	属性値の区切り
'	'	'	属性値の区切り（' は古い HTML で使えない）

数値文字参照の書き方

書き方	例	表示

進数;	あ	あ
進数;	あ	あ
&#x絵文字;	😀	😄

場所ごとのエスケープ

埋め込む場所	すること
HTML の本文	& < > をエスケープ
属性値 ("...")	& < > " ' をエスケープ。必ず引用符で囲む
URL (href)	URL エンコードしたうえで HTML エスケープ。javascript: は使わせない
<script> の中	HTML エスケープでは足りない。JSON.stringify して </ を <\/ に

よく使う文字参照

表示	書き方	表示	書き方
(空白)	 	©	©
®	®	™	™
¥	¥	€	€
×	×	÷	÷
...	…	–	–
←	←	→	→
↑	↑	↓	↓
«	«	»	»
½	½	°	°



文字コード表示 (Unicode・UTF-8・Shift_JIS)

チートシート

1文字ずつ、Unicode (U+3042)・UTF-8・Shift_JIS・EUC-JPのコードを表示。見えない文字の正体探しや、コードから文字に戻すことも。

「あ」のいろいろな書き方

書き方	「あ」	使う場所
Unicode	U+3042	仕様書・文字の説明
JavaScript / JSON / Java	\u3042	ソースコード・JSON
Python	\u3042 / chr(0x3042)	ソースコード
HTML 数値参照	あ / あ	HTML・XML
CSS	\3042	content: "\3042"
URLエンコード	%E3%81%82	URL のパラメーター
UTF-8 のバイト	E3 81 82	バイナリ・通信ログ
UTF-16	30 42	Windows の内部・Java
Shift_JIS	82 A0	Excel の CSV・古いシステム
EUC-JP	A4 A2	古い Linux・メール

見えない・まぎらわしい文字

文字	コード	注意
半角スペース	U+0020	
全角スペース	U+3000	見目で区別しにくい
ノーブレイクスペース	U+00A0	Web からコピーすると混ざる ()
ゼロ幅スペース	U+200B	見えない。パスワード・URL が通らない原因に
BOM	U+FEFF	ファイル先頭の見えない印。CSV の1列目名がずれる
改行 (Windows)	CR LF (0D 0A)	
改行 (Mac・Linux)	LF (0A)	
波ダッシュ ～	U+301C	Windows では ～ (U+FF5E) に化けることがある
全角マイナス -	U+FF0D	- (U+2212) と取り違えやすい

文字の種類ごとのバイト数

文字	UTF-8	Shift_JIS	UTF-16
半角英数字 A 1	1	1	2
半角カナ ア	3	1	2
ひらがな・漢字 あ 漢	3	2	2
①・(株) (環境依存)	3	2 (CP932)	2
・絵文字 😊	4	(使えない)	4 (サロゲートペア)



消費税計算（税込⇄税抜・インボイスの端数処理）

シートシート

税込⇄税抜を10%・8%で計算。品目を並べると、インボイスのルール（税率ごとに1回だけ端数処理）で請求書の合計を出します。

計算式

求めたいもの	10%	8%（軽減）
税抜 → 税込	税抜 × 1.1	税抜 × 1.08
税抜 → 消費税	税抜 × 0.1	税抜 × 0.08
税込 → 消費税	税込 × 10 ÷ 110	税込 × 8 ÷ 108
税込 → 税抜	税込 - 消費税	税込 - 消費税

早見表（端数切り捨て）

税抜	税込 10%	税込 8%
100円	110円	108円
500円	550円	540円
980円	1,078円	1,058円
1,000円	1,100円	1,080円
1,980円	2,178円	2,138円
3,000円	3,300円	3,240円
5,000円	5,500円	5,400円
10,000円	11,000円	10,800円
50,000円	55,000円	54,000円
100,000円	110,000円	108,000円

インボイス（適格請求書）のルール

項目	内容
端数処理	1枚の請求書につき、税率ごとに1回だけ
書くこと	登録番号・取引日・内容・税率ごとの合計と消費税額・宛名
軽減税率の印	「※」などを付けて「※は軽減税率対象」と書く
8% の対象	飲食料品（酒・外食を除く）、週2回以上発行の新聞の定期購読

Excel の端数処理

書き方	意味
<code>=ROUNDDOWN(A1*0.1,0)</code>	切り捨て
<code>=ROUND(A1*0.1,0)</code>	四捨五入
<code>=ROUNDUP(A1*0.1,0)</code>	切り上げ
<code>=ROUNDDOWN(A1*10/110,0)</code>	税込から消費税（切り捨て）



SQLの型変換の書き方（DB別早見表）

シートシート

CAST・CONVERT・TO_CHAR・TO_DATEの書き方を、MySQL・PostgreSQL・SQL Server・Oracle・SQLiteで比較。列名を入れてコピーできます。

文字列 → 数値

やりたいこと	MySQL	PostgreSQL	SQL Server	Oracle	SQLite
文字列を整数にする	CAST({col} AS SIGNED)	CAST({col} AS INTEGER)	CAST({col} AS INT)	TO_NUMBER({col})	CAST({col} AS INTEGER)
文字列を小数にする	CAST({col} AS DECIMAL(10,2))	CAST({col} AS NUMERIC(10,2))	CAST({col} AS DECIMAL(10,2))	TO_NUMBER({col})	CAST({col} AS REAL)
カンマ入りの文字列を数値にする	CAST(REPLACE({col}, ',', '') AS SIGNED)	CAST(REPLACE({col}, ',', '') AS INTEGER)	CAST(REPLACE({col}, ',', '') AS INT)	TO_NUMBER({col}, '999,999,999')	CAST(REPLACE({col}, ',', '') AS INTEGER)

数値 → 文字列

やりたいこと	MySQL	PostgreSQL	SQL Server	Oracle	SQLite
数値を文字列にする	CAST({n} AS CHAR)	CAST({n} AS TEXT)	CAST({n} AS VARCHAR(20))	TO_CHAR({n})	CAST({n} AS TEXT)
ゼロ埋めする（桁をそろえる）	LPAD({n}, 5, '0')	LPAD({n}::text, 5, '0')	RIGHT('00000' + CAST({n} AS VARCHAR(5)), 5)	LPAD({n}, 5, '0')	printf('%05d', {n})
3桁ごとにカンマを付ける	FORMAT({n}, 0)	TO_CHAR({n}, 'FM999,999,999')	FORMAT({n}, 'N0')	TO_CHAR({n}, 'FM999,999,999')	-- 標準関数では難しい（アプリ側での整形がおすすめ）
文字列と数値をつなげる	CONCAT('No.', {n})	'No.' {n}	CONCAT('No.', {n})	'No.' {n}	'No.' {n}

日付・日時

やりたいこと	MySQL	PostgreSQL	SQL Server	Oracle	SQLite
文字列を日付にする	STR_TO_DATE({s}, '%Y/%m/%d')	TO_DATE({s}, 'YYYY/MM/DD')	CONVERT(DATE, {s}, 111)	TO_DATE({s}, 'YYYY/MM/DD')	date(REPLACE({s}, '/', '-'))
文字列を日時にする	STR_TO_DATE({s}, '%Y-%m-%d %H:%i:%s')	TO_TIMESTAMP({s}, 'YYYY-MM-DD HH24:MI:SS')	CONVERT(DATETIME2, {s}, 120)	TO_TIMESTAMP({s}, 'YYYY-MM-DD HH24:MI:SS')	datetime({s})
日付を書式指定で文字列にする	DATE_FORMAT({d}, '%Y/%m/%d')	TO_CHAR({d}, 'YYYY/MM/DD')	FORMAT({d}, 'yyyy/MM/dd')	TO_CHAR({d}, 'YYYY/MM/DD')	strftime('%Y/%m/%d', {d})
日付を「2025年04月01日」にする	DATE_FORMAT({d}, '%Y年%m月%d日')	TO_CHAR({d}, 'YYYY"年"MM"月"DD"日"')	FORMAT({d}, 'yyyy年MM月dd日')	TO_CHAR({d}, 'YYYY"年"MM"月"DD"日"')	strftime('%Y年%m月%d日', {d})
日時から日付だけ取り出す（時刻を切り捨て）	DATE({dt})	{dt}::date	CAST({dt} AS DATE)	TRUNC({dt})	date({dt})
UNIX時間（秒）を日時にする	FROM_UNIXTIME({n})	TO_TIMESTAMP({n})	DATEADD(SECOND, {n}, '1970-01-01')	DATE '1970-01-01' + {n} / 86400	datetime({n}, 'unixepoch')
日時をUNIX時間（秒）にする	UNIX_TIMESTAMP({dt})	EXTRACT(EPOCH FROM {dt})	DATEDIFF(SECOND, '1970-01-01', {dt})	(CAST({dt} AS DATE) - DATE '1970-01-01') * 86400	strftime('%s', {dt})

計算・丸め

やりたいこと	MySQL	PostgreSQL	SQL Server	Oracle	SQLite
整数どうしの割り算で小数を出す	{n} / 2	{n} / 2.0	{n} * 1.0 / 2	{n} / 2	{n} * 1.0 / 2
四捨五入・切り捨て・切り上げ	ROUND({n}, 2)	ROUND({n}, 2)	ROUND({n}, 2)	ROUND({n}, 2)	ROUND({n}, 2)

NULL・エラー対策

やりたいこと	MySQL	PostgreSQL	SQL Server	Oracle	SQLite
NULLを別の値にする	IFNULL({col}, 0)	COALESCE({col}, 0)	ISNULL({col}, 0)	NVL({col}, 0)	IFNULL({col}, 0)
変換できない値はNULLにする（エラーにしない）	CASE WHEN {col} REGEXP '^?[0-9]+\$' THEN CAST({col} AS SIGNED) END	CASE WHEN {col} ~ '^?[0-9]+\$' THEN {col}::integer END	TRY_CAST({col} AS INT)	TO_NUMBER({col}) DEFAULT NULL ON CONVERSION ERROR)	CASE WHEN {col} GLOB '[0-9]*' AND {col} NOT GLOB '*[^\0-9]*' THEN CAST({col} AS INTEGER) END



ビジネスメール文例集（コピーして使える）

シートシート

お礼・お詫び・催促・お断り・日程調整など、名前や日付を入れてそのまま使えるメールの文例。

尊敬語・謙譲語

ふつう	尊敬語（相手が）	謙譲語（自分が）
言う	おっしゃる	申す・申し上げる
見る	ご覧になる	拝見する
行く	いらっしゃる	伺う・参る
来る	いらっしゃる・お越しになる	参る
いる	いらっしゃる	おる
する	なさる	いたす
食べる	召し上がる	いただく
知っている	ご存じだ	存じている・存じ上げる
聞く	お聞きになる	伺う・拝聴する
会う	お会いになる	お目にかかる
もらう	お受け取りになる	いただく・頂戴する
あげる	—	差し上げる
思う	お思いになる	存じる

メールの定型句

場所	例
書き出し	いつもお世話になっております。
初めて	突然のご連絡失礼いたします。
久しぶり	ご無沙汰しております。
お礼	ご対応いただき、ありがとうございます。
お詫び	ご迷惑をおかけし、申し訳ございません。
結び	引き続きよろしくお願いいたします。

間違えやすい敬語

NG	OK
おっしゃられる	おっしゃる
拝見させていただく	拝見する
ご覧になられる	ご覧になる
よろしかったでしょうか	よろしいでしょうか
了解しました（目上に）	承知しました
ご苦労様です（目上に）	お疲れ様です

クッション言葉

場面	言葉
お願いする	恐れ入りますが／お手数ですが
たずねる	差し支えなければ／失礼ですが
断る	あいにく／せっかくですが／残念ながら
反対する	おっしゃることは分かりますが
急ぎ	ご多忙のところ恐縮ですが



Windowsショートカットキー一覧

Win+V (クリップボード履歴)、Win+Shift+S (画面キャプチャ) など、パソコン作業が速くなるキー操作。覚えたものをチェックできます。

シートシート

基本 | 定番

Ctrl + C	★ コピー
Ctrl + X	★ 切り取り
Ctrl + V	★ 貼り付け
Ctrl + Z	★ 元に戻す
Ctrl + Y	やり直し (元に戻したのを取り消す)
Ctrl + A	★ すべて選択
Ctrl + S	★ 上書き保存
Ctrl + F	検索
Ctrl + P	印刷
Ctrl + N	新規作成 (新しいウィンドウ・ファイル)
Ctrl + マウスホイール	表示の拡大・縮小

基本 | ウィンドウ・アプリ

Alt + Tab	★ 開いているウィンドウを切り替え
Alt + F4	ウィンドウ・アプリを閉じる (デスクトップで押すとシャットダウン画面)
Ctrl + W	タブ・ファイルを閉じる (アプリは開いたまま)
Ctrl + Shift + Esc	★ タスクマネージャーを開く (固まったアプリの終了に)
Ctrl + Alt + Delete	ロック・サインアウト・タスクマネージャーの選択画面
Shift + F10	右クリックメニューを開く
F5	最新の情報に更新

Windowsキー | よく使う

Win + L	★ PCをロック (席を離れるときに)
Win + V	★ クリップボードの履歴から貼り付け (初回は有効化が必要)
Win + E	★ エクスプローラーを開く
Win + D	★ デスクトップを表示 (もう一度押すと元に戻る)
Win + R	★ ファイル名を指定して実行
Win + I	設定を開く
Win + X	クイックリンクメニュー (デバイスマネージャー・タミナルなど)
Win + . / Win + ;	絵文字・記号・顔文字の入力パネル
Win + H	音声入力
Win + 数字	タスクバーの左から○番目のアプリを起動・切り替え

Windowsキー | ウィンドウの配置

Win + ← / Win + →	★ ウィンドウの左半分・右半分にびったり配置
Win + ↑	ウィンドウを最大化
Win + ↓	元のサイズに最小化
Win + Shift + ← / Win + Shift + →	★ ウィンドウのモニターへ移動
Win + M	すべてのウィンドウを最小化
Win + Home	今のウィンドウを最小化
Win + Z	スナップレイアウト (Windows 11)
Win + P	★ プロジェクタや外部モニターへの映し方を切り替え

Windowsキー | 仮想デスクトップ

Win + Tab	タスクビュー (ウィンドウと仮想デスクトップの一覧)
Win + Ctrl + D	仮想デスクトップを新しく作る
Win + Ctrl + ← / Win + Ctrl + →	仮想デスクトップを切り替え
Win + Ctrl + F4	今の仮想デスクトップを閉じる

Windowsキー | そのほか

Win + A	クイック設定 (Wi-Fi・音量・Bluetooth)
Win + N	通知センター (Windows 11)
Win + K	ワイヤレスディスプレイに接続
Win + Ctrl + Shift + B	画面が真っ暗・固まったときに画面表示をリセット

エクスプローラー | ファイル操作

F2	★ 名前の変更
Ctrl + Shift + N	★ 新しいフォルダーを作る
Alt + Enter	プロパティを表示
Ctrl + Shift + C	★ ファイルのパスをコピー (Windows 11)
Shift + 右クリック	「パスのコピー」など、隠れたメニューを表示
Delete	ごみ箱に移動
Shift + Delete	ごみ箱に入れずに完全に削除 (元に戻せないので注意)
Ctrl + ドラッグ	コピーとしてドラッグ (同じドライブ内でも移動にしない)

エクスプローラー | 移動

Alt + ↑	★ 1つ上のフォルダーへ
Alt + ← / Alt + →	戻る・進む
Ctrl + L / Alt + D	★ アドレスバーを選択 (パスを入力・コピー)
Ctrl + E / Ctrl + F	検索ボックスへ
Ctrl + T	新しいタブで開く (Windows 11)

エクスプローラー | 表示

Alt + P	プレビューウィンドウの表示・非表示
Ctrl + Shift + G	詳細表示 (更新日時・サイズが見える)

文字入力・IME | IME (変換)

F6	ひらがなに変換
F7	★ 全角カタカナに変換
F8	半角カタカナに変換
F9	全角英数字に変換
F10	★ 半角英数字に変換 (日本語入力のまま英単語を打つときに)
変換	確定した文字を選んで押すと再変換
半角/全角	日本語入力のオン・オフ

文字入力・IME | カーソル移動・選択

Home / End	行の先頭・末尾へ移動
Ctrl + Home / Ctrl + End	文書の先頭・末尾へ移動
Ctrl + ← / Ctrl + →	単語単位で移動
Shift + 矢印	★ 選択範囲を広げる
Shift + Home / Shift + End	カーソルから行の先頭・末尾まで選択
Ctrl + Backspace	単語単位で削除
Ctrl + Shift + V	★ 書式なしで貼り付け (対応しているアプリで)

画面キャプチャ | スクリーンショット

Win + Shift + S	★ 範囲を選んでスクリーンショット (クリップボードへ)
PrtScn	画面全体をクリップボードへ (設定によっては切り取りツールが起動)
Alt + PrtScn	★ 今のウィンドウだけをクリップボードへ
Win + PrtScn	画面全体を「ピクチャ>スクリーンショット」に保存

画面キャプチャ | 録画

Win + Alt + R	画面の録画を開始・停止 (Xbox Game Bar)
Win + G	Xbox Game Barを開く



Officeショートカット一覧 (Excel・Word・PowerPoint・Outlook・Teams)

シートシート

セル内改行、今日の日付、形式を選択して貼り付け、全員に返信、ミュート切り替えなど、Officeの時短キーをアプリ別に。

Excel | 入力・編集

F2	★セルを編集モードにする（ダブルクリックと同じ）
Alt + Enter	★セル内で改行
Ctrl + ;	★今日の日付を入力
Ctrl + :	現在の時刻を入力
Ctrl + Enter	選択したすべてのセルに同じ値を入力
Ctrl + D	★上のセルをコピー（下方へ埋める）
Ctrl + R	左のセルをコピー（右方向へ埋める）
Ctrl + E	フラッシュフィル（入力例から規則を見つけて自動入力）
F4	★数式の中では参照の切り替え（\$A\$1 ⇄ A1）、それ以外では直前の操作を繰り返す
Ctrl + Alt + V	★形式を選択して貼り付け（値だけ貼るときに）
Alt + Shift + =	オートSUM（日本語キーボード。英語キーボードは Alt+=）
Shift + F2	メモ（コメント）を挿入・編集

Excel | 移動・選択

Ctrl + 矢印	★データの端まで一気に移動
Ctrl + Shift + 矢印	★データの端まで一気に選択
Ctrl + Home	A1セルへ移動
Ctrl + End	データがある最後のセルへ移動
Ctrl + Space	列全体を選択
Shift + Space	行全体を選択（日本語入力がおфのとき）
Ctrl + PageDown / Ctrl + PageUp	★次のシート・前のシートへ
Ctrl + G / F5	ジャンプ（セル番地を指定して移動、空白セルだけ選択など）

Excel | 行・列・シート

Ctrl + +	セル・行・列を挿入（テンキーの+。日本語キーボードは Ctrl+Shift+; でも可）
Ctrl + -	セル・行・列を削除
Ctrl + 9	行を非表示
Alt → H → O → I	列の幅を文字に合わせて自動調整
Shift + F11	新しいシートを挿入

Excel | 書式・表

Ctrl + 1	★セルの書式設定を開く
Ctrl + T	★テーブルに変換（見やすく、集計しやすくなる）
Ctrl + Shift + L	★フィルターのオン・オフ
Ctrl + B	太字
Ctrl + H	置換
F9	再計算（計算方法が手動のとき）
F12	名前を付けて保存

Word | 文字の書式

Ctrl + B / Ctrl + I / Ctrl + U	太字・斜体
Ctrl + J / Ctrl + I	フォントサイズ1ポイント大きく・小さく
Ctrl + Space	文字の書式をクリア（日本語入力がおфのとき）
Ctrl + Shift + C → Ctrl + Shift + V	★書式だけコピーして別かに貼り付け

Word | 段落

Ctrl + E	★中央揃え
Ctrl + L / Ctrl + R	左揃え・右揃え
Ctrl + J	両端揃え
Shift + Enter	段落を変えずに改行
Ctrl + Enter	★改ページ
Alt + Shift + ↑ / Alt + Shift + ↓	段落を上下に移動

Word | 校閲・編集

Ctrl + H	★置換
Ctrl + Shift + E	★変更履歴の記録のオン・オフ
Ctrl + Alt + M	コメントを挿入
F7	スペルチェックと文章校正
F4	直前の操作を繰り返す

PowerPoint | 作成

Ctrl + M	★新しいスライドを追加
Ctrl + D	★選んだ図形・スライドを複製
Ctrl + G	★図形をグループ化
Ctrl + Shift + G	グループ解除
Shift + ドラッグ	水平・垂直にまっすぐ移動、縦横比を保って拡大縮小
Ctrl + ドラッグ	図形をコピーしながら移動
Ctrl + Shift + C → Ctrl + Shift + V	図形の書式だけをコピーして貼り付け

PowerPoint | スライドショー

F5	★最初からスライドショー
Shift + F5	★今のスライドからスライドショー
Alt + F5	発表者ツールで開始
B / W	★画面を黒・白にする（もう一度押すと戻る）
数字 → Enter	その番号のスライドへジャンプ
Ctrl + L	レーザーポインター
Ctrl + P / E	ペンで書き込む・書き込みを消す
Esc	スライドショーを終了

Outlook | メール

Ctrl + R	★返信
Ctrl + Shift + R	★全員に返信
Ctrl + F	★転送（Outlook では検索ではないので注意）
Alt + S	送信
Ctrl + Shift + M	どの画面からでも新しいメールを作成
Ctrl + K	宛先の名前を確認（アドレス帳で補完）
Ctrl + Q / Ctrl + U	既読にする・未読にする
Ctrl + Shift + V	選んだメールを別のフォルダーへ移動

Outlook | 画面の切り替え

Ctrl + 1	メール
Ctrl + 2	★予定表
Ctrl + 3	連絡先
Ctrl + 4	タスク
Ctrl + E / F3	★検索ボックスへ

Teams | 会議

Ctrl + Shift + M	★マイクのミュート・解除
Ctrl + Shift + O	★カメラのオン・オフ
Ctrl + Shift + K	手を挙げる・下げる
Ctrl + Shift + E	画面の共有を開始
Ctrl + Space	ミュート中に押している間だけ話せる

Teams | チャット・移動

Ctrl + E	★検索
Ctrl + N	新しいチャット
Shift + Enter	★送信せずに改行
Ctrl + Shift + X	入力欄を広げる（書式付きで書く）
Ctrl + .	ショートカットの一覧を表示



ブラウザのショートカット一覧 (Chrome・Edge)

チートシート

閉じたタブを開き直す、キャッシュを無視して再読み込み、書式なし貼り付けなど、ChromeとEdgeのキー操作。

タブ・ウィンドウ | タブ

Ctrl + T	★ く
Ctrl + W	★ る
Ctrl + Shift + T	★ き じ
Ctrl + Tab / Ctrl + Shift + Tab	★ タ
Ctrl + 1~8	左 へ
Ctrl + 9	い
Ctrl + Shift + A	開 索

タブ・ウィンドウ | ウィンドウ

Ctrl + N	新しいウィンドウ
Ctrl + Shift + N	★ シークレット (Chrome) ・ InPrivate (Edge) ウィンドウ
F11	全画面表示

ページ操作 | 移動

Ctrl + L / Alt + D / F6	★ アドレ ーを選択
Alt + ← / Alt + →	★ 戻る・
Space / Shift + Space	1画面分下 上へスク ル
Home / End	ページの外 頭・末尾へ

ページ操作 | 表示

Ctrl + F	★
Ctrl + R / F5	再
Ctrl + Shift + R / Ctrl + F5	★ わ (に
Ctrl + + / Ctrl + - / Ctrl + 0	★ の
Ctrl + P	印 保
Ctrl + S	べ

リンク・マウス | リンク

Ctrl + クリック	★ リンク を新しいタ ブで開く (今のペー ジはそのま ま)
ホィールクリック	★ リンク を新しいタ ブで開く。 タブの上で 押すとタブ を閉じる
Ctrl + Shift + クリック	リンクを新 しいタブで 開いて、そ のタブに移 る
Shift + クリック	リンクを新 しいウィン ドウで開く
Alt + Enter	アドレスバ ーの内容を 新しいタブ で開く

リンク・マウス | 入力

Ctrl + Shift + V	★ 書式なしで貼 り付け (メー ル・Webフォー ムで文字だけ貼り たいとき)
------------------	--

履歴・ブックマーク | 履歴・ダウンロード

Ctrl + H	履歴を開く
Ctrl + J	★ ダウンロ ード一覧を 開く
Ctrl + Shift + Delete	閲覧履歴・ キャッシュ・Cookie の削除画面

履歴・ブックマーク | ブックマーク

Ctrl + D	★ 今のページをブ ックマーク (お気 入り) に追加
Ctrl + Shift + B	ブックマークバーの 表示・非表示
Ctrl + Shift + O	ブックマークの管理

開発者向け | 開発者ツール

F12 / Ctrl + Shift + I	★ 開発者ツ ールを開く
Ctrl + Shift + J	コンソールを
Ctrl + Shift + C	要素を選んで
Ctrl + U	ページのソー ス (HTML) を
Ctrl + Shift + M	開発者ツール の状態、フ ームに切り替

Edgeだけ | Microsoft Edge

Ctrl + Shift + S	★ Webキャプチャ (ページの一部や全 体を画像で保存)
Ctrl + Shift + U	音声で読み上げ
F9	イマーシブリーダー (広告を消して読み やすく表示)
Ctrl + Shift + ,	垂直タブの切り替え
Ctrl + Shift + Y	コレクションを開く



Markdown（マークダウン）の書き方・プレビュー

チートシート

左の書き方で書くと、右のように表示されます（GitHub・Qiita・Zenn・Backlog など共通の書き方）。

見出し・文字の装飾

書き方	表示
# 見出し1	見出し1
## 見出し2	見出し2
### 見出し3	見出し3
太字	太字
斜体	斜体
~~取り消し線~~	取り消し線
`コード`	コード
文の終わりに空白2つ 改行（または空行で段落）	文の終わりに空白2つ 改行（または空行で段落）

リスト・チェックボックス

書き方	表示
- りんご - みかん - 温州みかん	• りんご • みかん ◦ 温州みかん
1. 準備する 2. 実行する 3. 確認する	1. 準備する 2. 実行する 3. 確認する
- [x] 完了したタスク - [] まだのタスク	☒ 完了したタスク ☐ まだのタスク

リンク・画像・引用・区切り線

書き方	表示
[IT工具箱](https://example.com)	IT工具箱
<https://example.com>	https://example.com
> 引用した文章 ---	引用した文章

画像・脚注など（表示は書き方のとおり）

書き方	意味
![説明](image.png)	画像を表示
![![説明](image.png)](https://...)	画像にリンクを付ける
本文[^1] [^1]: 脚注の内容	脚注（GitHub・Qiita・Zenn）
*	記号をそのまま表示（エスケープ）
<!-- コメント -->	表示されないメモ

表

書き方	表示						
名前 点数 :--- ---: 山田 80 佐藤 95	<table><tr><th>名前</th><th>点数</th></tr><tr><td>山田</td><td>80</td></tr><tr><td>佐藤</td><td>95</td></tr></table>	名前	点数	山田	80	佐藤	95
名前	点数						
山田	80						
佐藤	95						
💡 :--- は左寄せ、---: は中央、---: は右寄せ							

コードブロック

書き方	表示
```js const a = 1; console.log(a); ```	<pre>const a = 1; console.log(a);</pre>
💡 の後ろに言語名（js・sql・python など）を書くと色が付きます	



# Windowsのよく使うコマンド集

チートシート

ファイル名を指定して実行 (Win+R) ・ コマンドプロンプト ・ PowerShell ・ Gitの定番コマンドを、用途から探してワンクリックでコピー。

## ファイル名を指定して実行 (Win+R) | 設定・管理画面

control	コントロールパネルを開く。
appwiz.cpl	「プログラムと機能」 (アプリのアンインストール画面) を開く。
ncpa.cpl	ネットワーク接続 (アダプターの設定) を開く。IPアドレスの手動設定に。
sysdm.cpl	システムのプロパティ (PC名・環境変数・リモートの設定) を開く。
devmgmt.ms c	デバイスマネージャーを開く。ドライバの確認に。
services.m sc	サービスの一覧を開く。
eventvwr.m sc	イベントビューアー (エラーの記録) を開く。
compmgmt.m sc	コンピューターの管理 (ディスク・ユーザー・共有フォルダー) を開く。
taskschd.m sc	タスクスケジューラ (定期実行) を開く。
msinfo32	システム情報 (CPU・メモリ・BIOS) を詳しく表示。
winver	Windows のバージョンとビルド番号を表示。
resmon	リソースモニター。どのアプリが通信・ディスクを使っているか分かる。
cleanmgr	ディスククリーンアップ (不要ファイルの削除)。

## ファイル名を指定して実行 (Win+R) | フォルダ

shell:startup	スタートアップフォルダを開く。ここに置いたショートカットはログイン時に自動で起動する。
shell:sendto	右クリックの「送る」のフォルダを開く。よく使うフォルダを登録できる。
%temp%	一時ファイルのフォルダを開く。容量の掃除に。
%appdata%	アプリの設定が保存されるフォルダ (AppData\Roaming) を開く。
shell:downloa ds	ダウンロードフォルダを開く。
\\サーバー名 \共有名	ネットワーク上の共有フォルダを開く。

## ファイル名を指定して実行 (Win+R) | アプリ

cmd	コマンドプロンプトを起動。Ctrl+Shift+Enter で管理者として起動。
powershe ll	PowerShell を起動。
notepad	メモ帳を起動。
calc	電卓を起動。
snipping tool	切り取り & スケッチ (画面キャプチャ) を起動。Win+Shift+S でも可。
mstsc	リモートデスクトップ接続を起動。
excel /safe	Excel をセーフモードで起動 (アドインが原因の不具合の切り分けに)。winword /safe で Word も。
outlook /safe	Outlook をセーフモードで起動。

## コマンドプロンプト | システム・プロセス

tasklist	実行中のプロセス一覧。
taskkill /PID 番号 /F	⚠ プロセスを強制終了。/IM でプログラム名指定も。
systeminfo	OSのバージョン・メモリ・起動時刻などを表示。
hostname	PC名を表示。
whoami	ログイン中のユーザー名を表示。
shutdown /r /t 0	⚠ すぐに再起動。/s でシャットダウン。
sfc /scannow	システムファイルの破損を検査・修復 (管理者で実行)。
chcp 65001	文字コードをUTF-8に切り替え (文字化け対策)。932 でShift_JISに戻す。
set	環境変数を表示・設定 (そのウィンドウ内だけ)。永続化は setx。
cls	画面をクリア。
gpupdate /force	グループポリシー (会社の設定) をすぐに反映させる。
powercfg /batteryreport	ノートPCのバッテリーの劣化具合をレポート (HTML) に出力。
winget upgrade -- all	インストール済みアプリをまとめて更新 (Windows 10/11 の winget)。
driverquery	インストールされているドライバーの一覧。
schtasks /query /fo LIST	登録されているスケジュールタスクの一覧。

## PowerShell | ファイル・フォルダ

Get-ChildItem -Recurse -Filter *.log	ファイルを再帰的に一覧 (ls / dir と同じ)。
Copy-Item 元 先 -Recurse	フォルダを中身ごとコピー。
Remove-Item パス -Recurse -Force	⚠ フォルダを中身ごと削除。まず -WhatIf を付けて対象を確認するのが安全。
Get-ChildItem *.txt   Rename-Item - NewName { \$_.Name -replace "old","new" }	ファイル名を一括で置換してリネーム。
Compress-Archive -Path フォルダ - DestinationPath out.zip	ZIPに圧縮。Expand-Archive で展開。
Get-FileHash ファイル	ファイルのハッシュ値 (SHA256) を表示。ダウンロードファイルの改ざん確認に。
Get-ChildItem -Recurse   Where-Object LastWriteTime -gt (Get- Date).AddDays(-7)	最近7日以内に更新されたファイルを探す。
Get-ChildItem   Select-Object Name   Export-Csv list.csv -NoTypeInfoation -Encoding UTF8	フォルダ内のファイル名一覧をCSVに書き出す (Excelで一覧表を作るときに)。

コマンドプロンプト   ファイル・フォルダ	
dir	フォルダの中身を一覧表示。/s でサブフォルダも、/b で名前だけ。
cd フォルダ	フォルダを移動。別ドライブへ移るときは /d を付ける。
mkdir フォルダ名	フォルダを作成 (md でも同じ)。途中のフォルダもまとめて作れる。
copy 元 先	ファイルをコピー。
robocopy 元 先 /E	フォルダを中身ごとコピー。大量ファイルや再実行に強い。/MIR はミラー（先にしか無いファイルは削除）なので注意。
move 元 先	ファイルの移動・名前変更。
ren 旧名 新名	ファイル名を変更。ワイルドカードで拡張子の一括変更もできる。
del ファイル	 ファイルを削除（ごみ箱に入らない）。/s でサブフォルダも。
rmdir /s /q フォルダ	 フォルダを中身ごと確認なしで削除。
tree /f	フォルダ構成をツリー表示。/f でファイルも。> で保存すると資料づくりに便利。
type ファイル	テキストファイルの中身を表示。
start .	今いるフォルダをエクスプローラーで開く。
xcopy 元 先 /E /I /Y	フォルダを中身ごとコピー（robocopy が使えない古い環境向け）。
attrib -h -s ファイル	隠しファイル・システムファイルの属性を外す。
certutil -hashfile ファイル SHA256	ファイルのハッシュ値を表示（ダウンロードしたファイルの確認に）。

コマンドプロンプト   検索・テキスト	
findstr "文字列" ファイル	ファイルの中から文字列を検索（grep 相当）。/s でサブフォルダ、/i で大文字小文字を区別しない。
where コマンド名	コマンドの実体はどこにあるかを表示。
fc ファイル1 ファイル2	2つのファイルの差分を表示。
dir /s /b ファイル名	ファイルをサブフォルダまで探して、見つかったパスを表示。
コマンド   clip	コマンドの結果をクリップボードにコピー。

PowerShell   テキスト・CSV	
Get-Content ファイル -Tail 50 -Wait	ログの末尾を表示し、追記を監視（tail -f 相当）。
Select-String -Path *.log -Pattern "error"	複数ファイルから文字列を検索（grep 相当）。
Import-Csv in.csv   Export-Csv out.csv -NoTypeInfoation -Encoding UTF8	CSVを読み込んで書き出す（文字コード変換や列の加工に）。
Get-Content in.txt -Encoding Default   Set-Content out.txt -Encoding UTF8	Shift_JIS のファイルを UTF-8 に変換（Windows PowerShell 5.1）。
(Get-Content ファイル).Count	ファイルの行数を数える。
Get-Clipboard / Set-Clipboard	クリップボードの読み書き。

PowerShell   システム・プロセス	
Get-Process   Sort-Object CPU -Descending   Select-Object -First 10	CPUを多く使っているプロセス上位10件。
Stop-Process -Name プロセス名	 プロセスを終了。
Get-Service   Where-Object Status -eq Running	実行中のサービス一覧。
Set-ExecutionPolicy RemoteSigned -Scope CurrentUser	スクリプト（.ps1）を実行できるようにする。
\$PSVersionTable	PowerShell のバージョンを表示。
Get-History	このセッションで実行したコマンドの履歴。
Get-CimInstance Win32_OperatingSystem   Select-Object LastBootUpTime	最後に起動した日時（再起動したかの確認に）。
Get-ComputerInfo   Select-Object OsName, OsVersion, CsTotalPhysicalMemory	OS名・バージョン・メモリ容量を表示。
Restart-Computer	 PCを再起動。

PowerShell   ネットワーク	
Test-NetConnection ホスト -Port 443	特定ポートにつながるか確認（ping では分からないとき）。
Invoke-WebRequest URL -OutFile ファイル	ファイルをダウンロード。
Invoke-RestMethod URL	APIを呼んでJSONをオブジェクトとして受け取る。
Get-NetIPAddress -AddressFamily IPv4	IPv4 アドレスの一覧。

Git（Git for Windows）   基本	
git status	変更されたファイル・ステージ状況を確認。
git add -p	変更を部分ごとに確認しながらステージ。
git commit -m "メッセージ"	コミット。
git commit --amend	直前のコミットを修正（push 済みなら原則使わない）。
git log --oneline --graph -all	ブランチの分岐を含めて履歴をコンパクトに表示。
git diff --staged	ステージ済みの変更を表示（コミット前の確認に）。

Git（Git for Windows）   ブランチ	
git switch -c ブランチ名	ブランチを作って切り替え（git checkout -b と同じ）。
git branch -d ブランチ名	マージ済みのブランチを削除。
git merge ブランチ名	指定したブランチを今のブランチに取り込む。
git rebase main	今のブランチを main の先頭に付け替える。

コマンドプロンプト   ネットワーク	
ipconfig	IPアドレス・デフォルトゲートウェイを表示。/all でMACアドレスやDNSも。
ipconfig /flushdns	DNSキャッシュを消去。サイトの切り替え直後につながらないときに。
ping ホスト	相手に届くか確認。-t で止めるまで送り続ける。
tracert ホスト	相手までの経路（どこで止まっているか）を調べる。
nslookup ドメイン	ドメインのIPアドレス（DNS）を調べる。
netstat -ano	通信中・待ち受け中のポートとプロセスIDを表示。
curl URL	URLにアクセスして中身を表示（Windows 10 以降は標準搭載）。-I でヘッダーだけ。
ipconfig /release && ipconfig /renew	IPアドレスを取り直す。ネットにつながらないときの定番。
getmac /v	MACアドレスを一覧表示。
arp -a	同じネットワーク内で通信した機器のIPとMACアドレスの一覧。
netsh wlan show profiles	PCに保存されているWi-Fiの一覧。
net use Z: \\サーバー\\共有	共有フォルダをネットワークドライブ（Z:）として割り当てる。解除は net use Z: /delete。

Git（Git for Windows）   リモート	
git pull --rebase	リモートの変更を取り込む（マージコミットを作らない）。
git push -u origin ブランチ名	新しいブランチを初めて push。
git fetch --prune	リモートで消えたブランチの参照も掃除しつつ取得。

Git（Git for Windows）   やり直し	
git restore ファイル	⚠ 作業中の変更を取り消す（最後のコミットの状態に戻す）。
git restore --staged ファイル	add を取り消す（変更そのものは残る）。
git reset --soft HEAD~1	直前のコミットを取り消す（変更は残る）。
git revert コミットID	指定したコミットを打ち消すコミットを作る（push 済みでも安全）。
git stash	作業中の変更を一時退避。git stash pop で戻す。
git reflog	HEAD の移動履歴。消したと思ったコミットを取り戻すときに。

Git（Git for Windows）   調査	
git blame ファイル	各行を最後に変更したコミットと人を表示。
git log -S "文字列"	その文字列を追加・削除したコミットを探す。
git config --global core.quotePath false	日本語ファイル名が \346\227... のように表示されるのを直す。