



CURSO BÁSICO DE R COM ECONOMETICA

Projeto contabiliDados

Curso de Ciências Contábeis - Área de Finanças

PROJETO CONTABILIDADOS

Esse projeto se justifica pela oportunidade de sumarizar os dados públicos de entidades públicas e privadas para a sociedade em geral, levando informação de qualidade para acompanhamento da gestão. O simples fato de divulgar não torna as informações disponíveis relevantes para a sociedade. A partir do momento que se aproveita os preceitos de governança corporativa estimuladas pela Comissão de Valores Mobiliários (CVM), e as determinações legais advindas da Lei Complementar nº 101 (2000), popularmente conhecida como Lei de Responsabilidade Fiscal (LRF), da Lei Complementar nº 131 (2009) que alterou a LRF no que concerne a transparência, combinadas com a Lei nº 12.527 (2011) – Lei de Acesso à Informação (LAI), que versa sobre o direito do cidadão ao acesso à informação, é possível tornar os dados dispersos em informações úteis aos cidadãos interessados nas finanças do Brasil.

Esse projeto, consiste, portanto, em acompanhar o desempenho e atividades de entidades públicas e privadas, por meio de consolidação de dados divulgados de forma dispersa, trazendo para a sociedade indicadores sumarizados sobre a gestão pública e corporativa no Brasil. Ao mesmo tempo cuida em manter bancos de dados disponíveis para futuras pesquisas de interesse público e privado com divulgação de índices de sentimento e indicadores.

OBJETIVOS

- Mapear bases de dados disponíveis na web;
- Organizar a base central de dados orçamentários, financeiros, contábeis e de gestão de entidades públicas e privadas;
- Criar rotina para tratamento dos dados;
- Disponibilizar dados à comunidade;
- Treinar docentes, discentes e membros da sociedade em geral com interesse em fazer uso da ciência de dados para avaliar a gestão pública e privada no Brasil.

METODOLOGIA

Após mapear as fontes de dados disponíveis e confiáveis, serão direcionados os tipos de dados a serem explorados (financeiros, fiscais, previdenciários, gestão, dentre outros). Com base na definição dos dados a serem explorados será definida a política do observatório, iniciando com a criação de rotinas de extração, tratamento e divulgação dos dados.

Com a compreensão das rotinas e dados a serem explorados, serão definidas as equipes para estruturação das bases de dados, tendo como objetivo inicial a sumarização e “clusterização” dos dados. Para fins de benchmark de coleta estão repositórios públicos como por exemplo, na Secretaria do Tesouro Nacional, IBGE, Federação das Indústrias do Rio de Janeiro (FIRJAN), Fundo Nacional de Educação, Fundo Nacional da Saúde, RAIS, CAGED, Comissão de Valores Mobiliários (CVM), Brasil Bolsa Balcão (B3), entre outras.

A etapa seguinte consiste em calcular indicadores e/ou índices de gestão, contábil ou de sentimento, de forma a produzir informações individualizadas de entidades públicas e privadas, definindo o formato de divulgação dos dados (sumarizados ou microdados). Para essa etapa serão utilizados diversos softwares, tais como: R, Stata, Matlab, Power BI, Python e Excel. Nessa etapa se iniciam os treinamentos de alunos e professores para condução dos subprojetos, divididos por área de interesse a ser definida após tratamento dos dados. Ao final serão construídos os dicionários das bases de dados criadas e definidos os painéis para consulta de dados (dashboards) com o respectivo treinamento de alunos, professores e sociedade em geral.

Mais informações sobre o Projeto contabiliDados **clique aqui**

SOBRE ESTE TREINAMENTO

EMENTA

Apresentação da plataforma Economatica. Utilização das ferramentas screener e matrix. Geração de bancos de dados em excel em séries temporais e dados em painel. Apresentação da plataforma R e R Studio. Operações básicas no R. Tipos de objeto. Importação de planilhas e dados externos. Geração de banco de dados. Estatística descrita com valores e gráficos.

OBJETIVOS

- Estimular os alunos a utilizar ferramentas de análise de dados em atividades relacionadas à área contábil;
- Capacitar os alunos na ferramenta R Programming;
- Abordar, na prática, conceitos de contabilidade, estatística e análise de dados.

METODOLOGIA

Serão utilizados dados da plataforma Economatica para geração de indicadores contábeis básicos com o intuito de demonstrar usos básicos da plataforma R Programming. Os dados gerados serão importados, tratados e analisados, propiciando a interação entre os alunos e as ferramentas estudadas.

MATERIAL COMPLEMENTAR

A apresentação desse treinamento pode ser acessada pelo canal do youtube do Projeto UFERSA3 - Liga de Mercado Financeiro da UFERSA:

Parte 1: [clique aqui](#)

Parte 2: [clique aqui](#)

O canal do youtube também dispõe de vídeos com exploração de ferramentas básicas e avançadas do Economatica.

Acesse: www.youtube.com/c/ProjetoUfersa3

EQUIPE RESPONSÁVEL

Prof. **Kléber Formiga** Miranda

E-mail - mirandakf@ufersa.edu.br | **Lattes** - lattes.cnpq.br/5969359253746807

Github - github.com/kleberformiga | **Linkedin** - br.linkedin.com/in/kleberformiga

Prof. **Lucas Lúcio Godeiro**

E-mail - lucassgodeiro@ufersa.edu.br | **Lattes** - lattes.cnpq.br/0903970970268664

Github - github.com/lucassgodeiro | **Linkedin** - br.linkedin.com/in/lucas-godeiro

Prof. **Alexsandro** Gonçalves da Silva **Prado**

E-mail - alexsandro.prado@ufersa.edu.br | **Lattes** - lattes.cnpq.br/4254857944224293

Github - github.com/alexsandroprado | **Linkedin** - br.linkedin.com/in/alexsandroprado

Kewerson Alves **Cunha**

E-mail - kewerson.cunha@alunos.ufersa.edu.br | **Linkedin** - br.linkedin.com/in/kewerson-alves-cunha

Tomaz da Silva **Melo**

E-mail - tomaz.melo@alunos.ufersa.edu.br | **Lattes** - lattes.cnpq.br/5153159638802664

Github - github.com/TomazMelo | **Linkedin** - br.linkedin.com/in/tomaz-melo

Isabele Vieira de **Matos**

E-mail - isabele.matos@alunos.ufersa.edu.br | **Lattes** - lattes.cnpq.br/3188106686594869

Linkedin - br.linkedin.com/in/isabele-matos

ECONOMATICA

A Economatica é constantemente alimentada com os dados mais recentes do mercado financeiro e permite ao usuário manipular um grande volume de dados, encontrá-los facilmente e capturá-los sem estresse para fazer simulações avançadas e comparar ativos com maior facilidade e eficiência. Tudo isso, para oferecer uma visão mais ampla e melhor subsídio para produzir melhores análises, realizar melhores relatórios e obter melhores insights, enquanto economiza tempo de suas equipes. (Fonte: Site Economatica, 2022).

Neste treinamento, vamos mostrar como acessar o economatica para realizar uma tarefa simples: coletar o Ativo Circulante (AC) e o Passivo Circulante (PC) de empresas do setor elétrico para cálculo da liquidez corrente (AC/PC). A proposta é demonstrar o passo a passo para coletar dados contábeis na plataforma Economatica, gerando um banco de dados em excel com arquivo no formato CSV, em formato de painel.

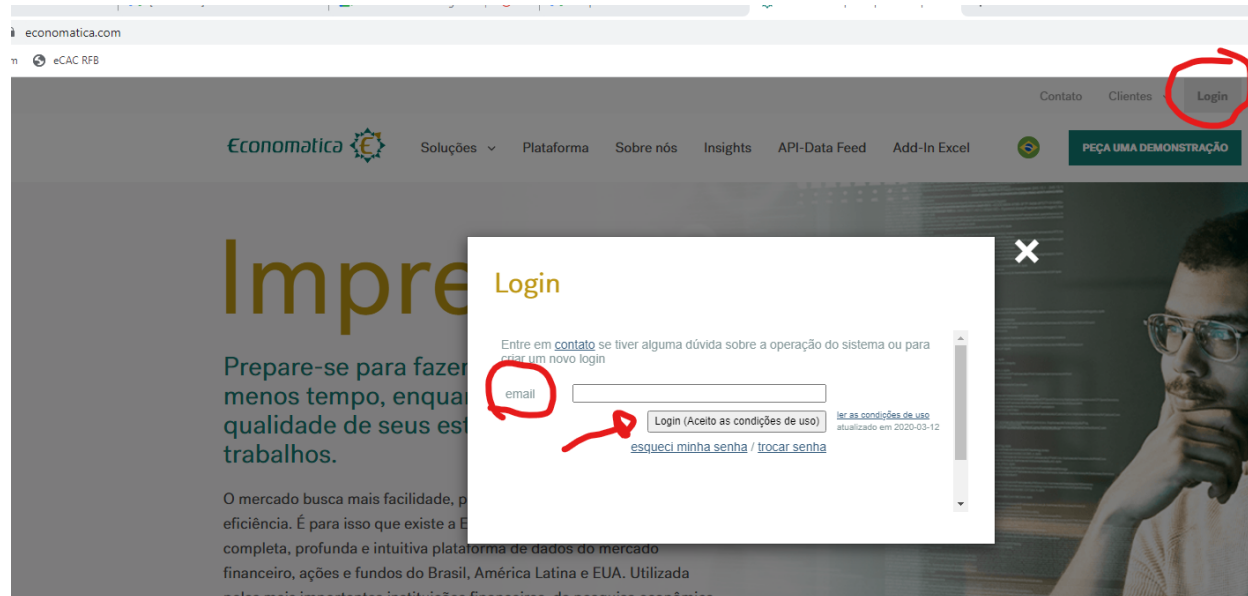
O formato painel é comumente usado em pesquisas da área contábil. Essencialmente, dispor os dados em painel significa que teremos uma variável (coluna) com o código da empresa (id ou indivíduo), outra com o tempo (t) e as demais variáveis de interesse. No caso desse treinamento, o ativo circulante e o passivo circulante. Os dados ficarão assim:

cod	ano	ativoCirc	passivoCirc
ABCD3	2018	123.415	100.280
ABCD3	2019	100.420	104.718
ABCD3	2020	101.213	99.530
XYKW3	2018	52.400	40.313
XYKW3	2019	77.681	76.990
XYKW3	2020	43.520	49.333

Observe que as empresas ABCD3 e XYKW3 aparecem empilhadas na coluna cod e possuem os anos de 2018 a 2020 na coluna ano, seguidos dos valores das variáveis de interesse: Ativo Circulante (ativoCirc) e Passivo Circulante (passivoCirc). Essa disposição de dados em uma tabela é denominada “Dados em Painel.” Embora existam outras disposições para esses dados, escolhemos esse formato por ser mais usual.

Coletando os dados

Passo 1: Acessar a plataforma Economatica



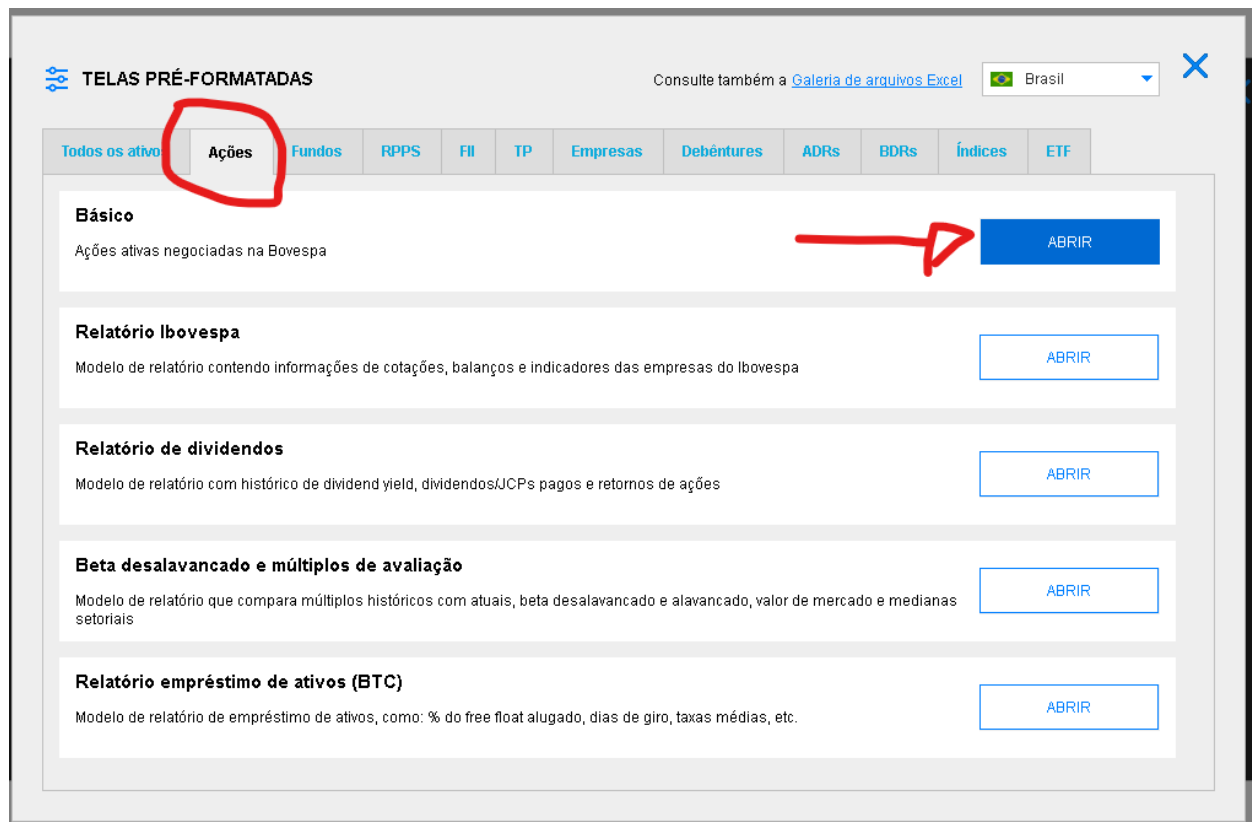
Conforme pode ser visto na figura acima, após o usuário acessar o site www.economatica.com, clica em [Login](#) para que apareça uma tela com o email e o botão para realizar o login. O email informado deverá ser o do usuário para controle de acesso. O acesso é livre dentro das dependências da UFERSA. Pelo contrato vigente, qualquer máquina dentro do IP da UFERSA terá acesso ao economatica (limite de 50 acessos simultâneos).

Passo 2: Tela Inicial



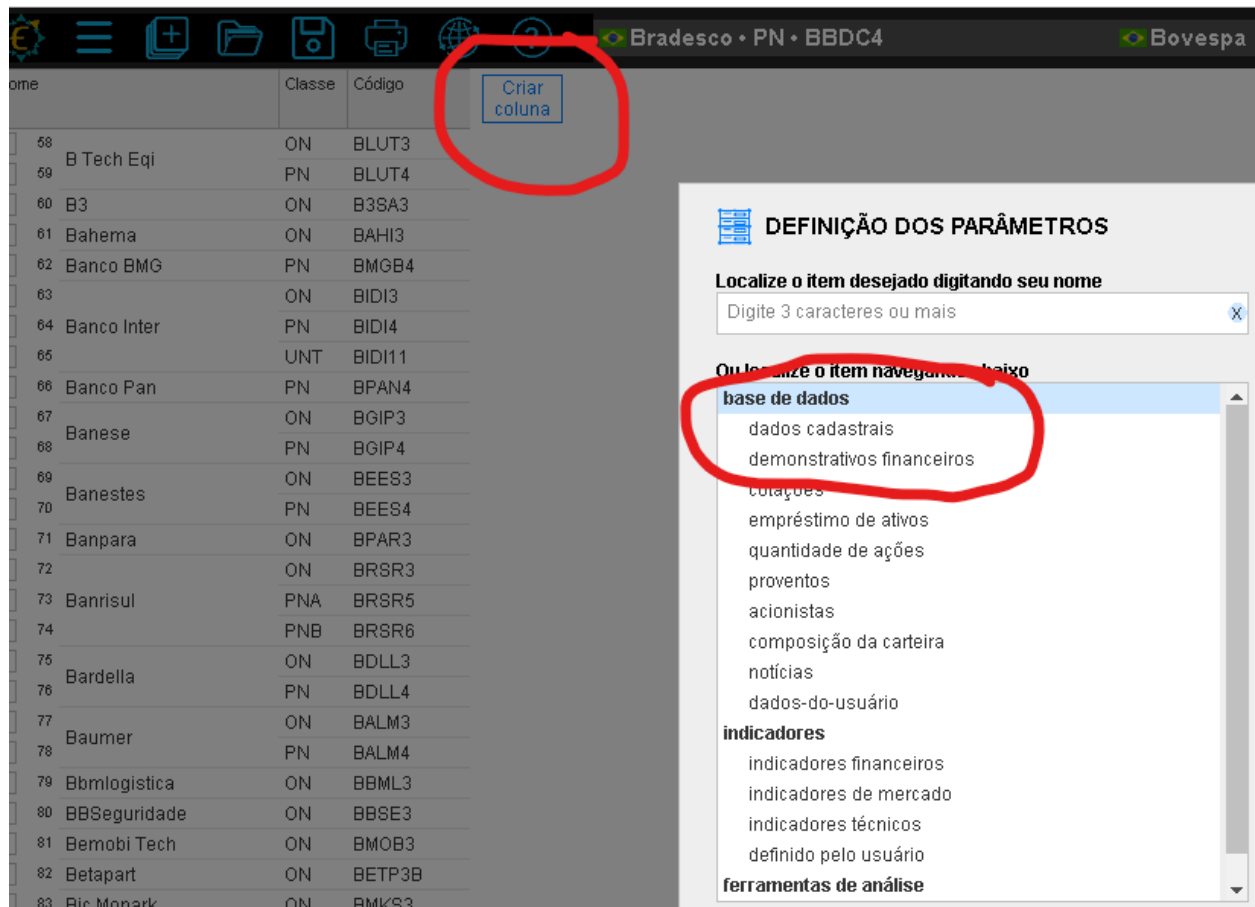
Após login, aparecerá a tela inicial. Neste treinamento, vamos nos concentrar em duas opções: Screening e Matrix (ver destaque). Vamos iniciar pelo Screening.

Passo 3: Screening - Escolha da Tela Pré-Formatada



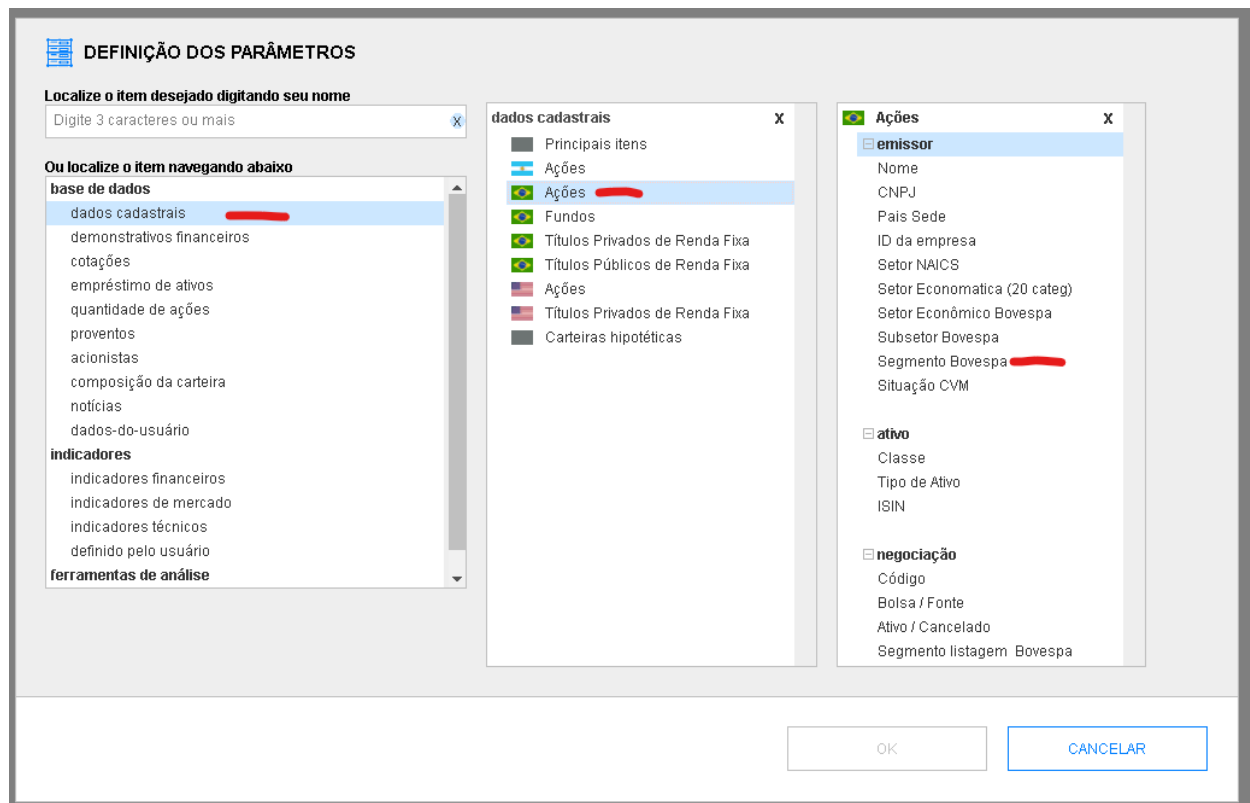
O Economatica dispõe de várias de várias telas pré-formatadas. Vamos usar o screening pré-formatado “Básico.” Nesse screening serão selecionadas as empresas que negociam ações na B3 (Brasil, Bolsa, Balcão) e que estejam ativas. Basta escolher a aba “Ações” e clicar no botão “ABRIR,” conforme figura acima.

Passo 4: Screening - Escolha de dados



Após aberto o screening, clica na opção “Criar Coluna” para que apareça a janela com várias opções. Vamos usar a opção “Dados Cadastrais.”

Passo 5: Screening - Selecciona Segmento



Clica em “Dados Cadastrais” -> “Ações ->”Segmento Bovespa”.

Com essa sequência, será incluída uma coluna “Segmento Bovespa” na qual constarão o segmento de cada empresa selecionada no screening básico, conforme figura a seguir.

Passo 6: Screening - Filtra Segmento

The screenshot shows the Bradesco PN BBDC4 interface. A table lists companies with columns: Nome, Classe, Código, and Segmento Bovespa. A right-click context menu is open over the 'Segmento Bovespa' column header, showing options like 'Alterar parâmetros desta coluna', 'Repetir esta coluna com outros parâmetros', 'Inserir coluna à esquerda desta', 'Ocultar coluna', 'Apagar coluna', 'Crescente', 'Decrescente', 'Inserir filtro baseado nesta coluna', 'Agrupar por essa coluna', 'Marcar os itens selecionados', 'Selecionar tudo', and 'Exportar para Excel Add-In'. The 'Inserir filtro baseado nesta coluna' option is highlighted with a red line.

Nome	Classe	Código	Segmento Bovespa
58 B Tech Eqi	ON	BLUT3	Outr
59	PN	BLUT4	Outr
60 B3	ON	B3SA3	Serv
61 Bahema	ON	BAHI3	Serv
62 Banco BMG	PN	BMGB4	Banc
63	ON	BIDI3	Banc
64 Banco Inter	PN	BIDI4	Banc
65	UNT	BIDI11	Banc
66 Banco Pan	PN	BPAN4	Banc
67 Banese	ON	BGIP3	Banc
68	PN	BGIP4	Banc
69 Banestes	ON	BEE33	Banc
70	PN	BEE34	Banc
71 Banpara	ON	BPAP3	Bancos
72	ON	BR3R3	Bancos
73 Banrisul	PNA	BR3R5	Bancos
74	PNB	BR3R6	Bancos
75 Bardella	ON	BDLL3	Máq. e equip....
76	PN	BDLL4	Máq. e equip....

Como a proposta é estudar empresas do setor elétrico, vamos filtrar apenas empresas do segmento “Energia Elétrica.” Para isso, basta clicar com o botão direito do mouse no título da coluna “Segmento Bovespa” e escolher a opção do menu “Inserir filtro baseado nesta coluna.” Aparecerá a seguinte tela:

The screenshot shows the Bradesco PN BBDC4 interface with the 'FILTROS' (Filters) section. The 'Segmento Bovespa' filter is selected. A dialog box titled 'SEGMENTO BOVESPA' is open, showing a list of sectors. The 'Energia elétrica' option is selected and highlighted with a red circle. The 'Mostrar apenas uma classe por empresa (a de maior volume no último mês)' checkbox is also checked and highlighted with a red circle.

FILTROS

- Bolsa / Fonte: Igual
- Tipo de Ativo: Igual
- Ativo / Cancelado: Igual
- Segmento Bovespa: Igual

☒ Mostrar apenas uma classe por empresa (a de maior volume no último mês)

SEGMENTO BOVESPA

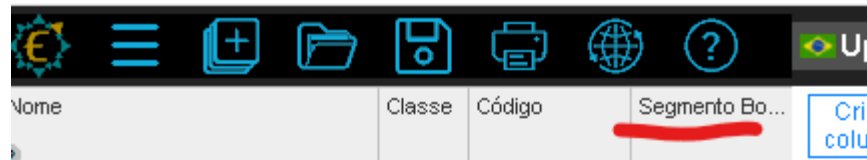
- ☐ Corretoras de seguros
- ☐ Eletrodomésticos
- ☐ Embalagens
- ☒ Energia elétrica
- ☐ Engenharia consultiva
- ☐ Equipamentos
- ☐ Equipamentos e serviços
- ☐ Equipamentos elétricos
- ☐ Exploração de imóveis
- ☐ Exploração de rodovias
- ☐ Exploração refino e distribuição
- ☐ Fertilizantes e defensivos
- ☐ Fios e tecidos
- ☐ Gás
- ☐ Gestão de recursos e investimentos
- ☐ Holdings diversificadas
- ☐ Interalia

O primeiro procedimento nesta tela é clicar na opção “Mostrar apenas uma classe por empresa (a de maior volume no último mês).” Esta opção faz com que seja apresentada apenas um código por empresa (o de

maior de negociação no mês anterior). Como estamos coletando dados contábeis, essa opção faz sentido, já que o ativo circulante da PETR3 é o mesmo que o da PETR4 em cada ano. Essa opção não faz sentido quando coletarmos dados referente a negociações, como preço e volume, pois cada papel (PETR3 e PETR4, por exemplo) tem valores diferentes, diariamente. Assim, selecionaremos essa opção (basta clicar).

O segundo procedimento é abrir a listBox do filtro da coluna “Segmento Bovespa.” Após clicar no listBox, aparecerá a janela da figura acima. Selecionaremos o segmento “Energia Elétrica.”

Após clicar “OK,” o screening terá apenas um código por empresas e apenas as empresas do segmento “Energia Elétrica,” conforme figura abaixo:



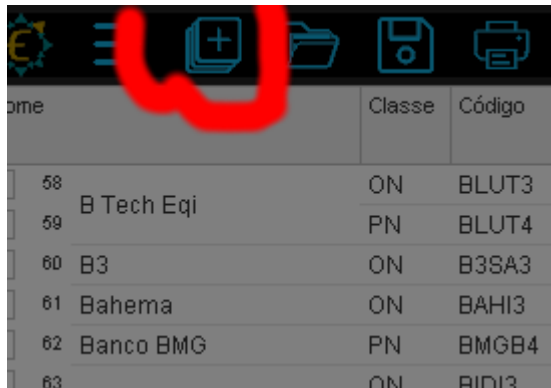
	Nome	Classe	Código	Segmento Bovespa
<input type="checkbox"/>	4 Alupar	UNT N2	ALUP11	Energia elétrica
<input type="checkbox"/>	5 Ampla Energ	ON	CBEE3	Energia elétrica
<input type="checkbox"/>	6 Auren	ON	AURE3	Energia elétrica
<input type="checkbox"/>	7 Ceb	PNB	CEBR6	Energia elétrica
<input type="checkbox"/>	8 Ceee-D	ON	CEED3	Energia elétrica
<input type="checkbox"/>	9 Ceee-T	PN	EEEL4	Energia elétrica
<input type="checkbox"/>	10 Celesc	PN	CLSC4	Energia elétrica
<input type="checkbox"/>	11 Celgpar	ON	GPAR3	Energia elétrica
<input type="checkbox"/>	12 Celpe	PNA	CEPE5	Energia elétrica
<input type="checkbox"/>	13 Cemig	PN	CMIG4	Energia elétrica
<input type="checkbox"/>	14 Cesp	PNB	CESP6	Energia elétrica
<input type="checkbox"/>	15 Coelba	ON	CEEB3	Energia elétrica
<input type="checkbox"/>	16 Coelce	PNA	COCE5	Energia elétrica
<input type="checkbox"/>	17 Comerc	ON	COMR3	Energia elétrica
<input type="checkbox"/>	18 Copel	PNB	CPLE6	Energia elétrica
<input type="checkbox"/>	19 Cosern	ON	CSRN3	Energia elétrica
<input type="checkbox"/>	20 CPFL Energia	ON	CPFE3	Energia elétrica
<input type="checkbox"/>	21 Elektro	PN	EKTR4	Energia elétrica
<input type="checkbox"/>	22 Eletrobras	ON	ELET3	Energia elétrica
<input type="checkbox"/>	23 Eletropar	ON	LIPR3	Energia elétrica
<input type="checkbox"/>	24 Emae	PN	EMAE4	Energia elétrica
<input type="checkbox"/>	25 Energias BR	ON	ENBR3	Energia elétrica
<input type="checkbox"/>	26 Energisa	UNT N2	ENGI11	Energia elétrica
<input type="checkbox"/>	27 Energisa Mt	ON	ENMT3	Energia elétrica
<input type="checkbox"/>	28 Eneva	ON	ENEV3	Energia elétrica
<input type="checkbox"/>	29 Engie Brasil	ON	EGIE3	Energia elétrica
<input type="checkbox"/>	30 EqtI Maranhao	ON	EQMA3B	Energia elétrica
<input type="checkbox"/>	31 EqtI Para	ON	EQPA3	Energia elétrica
<input type="checkbox"/>	32 Equatorial	ON	EQTL3	Energia elétrica
<input type="checkbox"/>	33 Ger Paranap	PN	GEPA4	Energia elétrica

É possível observar a lista de empresas apenas do segmento selecionado e apenas uma classe por empresa.

Temos, portanto, a lista de empresas definidas para este treinamento: empresas do setor elétrico.

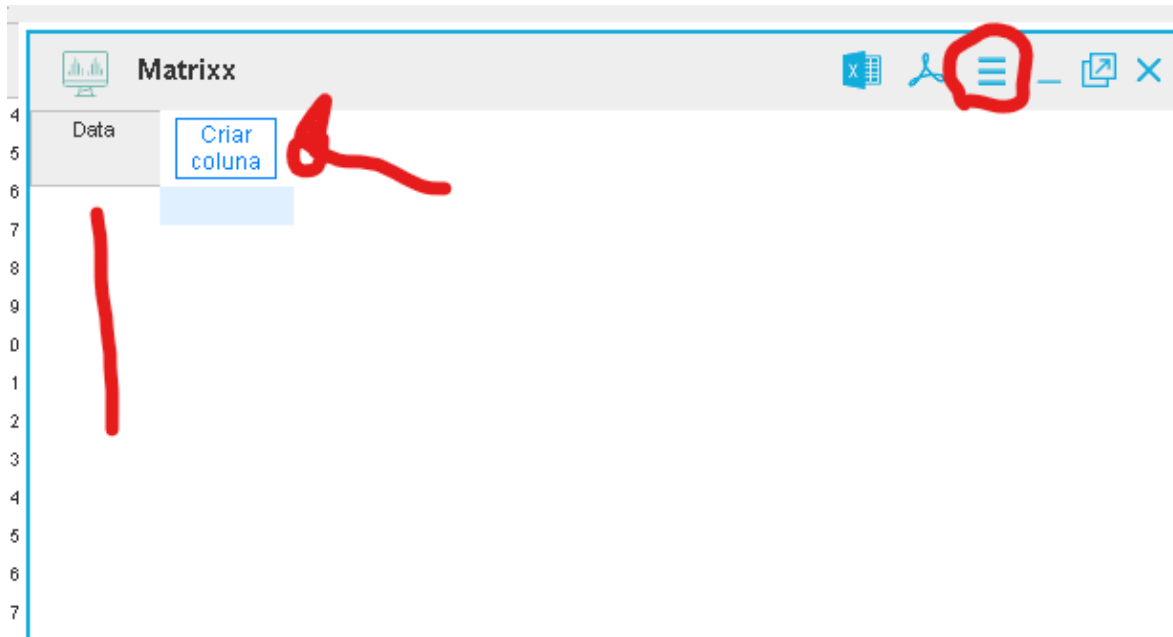
Passo 7: Matrix - Estrutura

Sem fechar o screening, escolha outro serviço clicando no botão com sinal de “+” indicado na figura abaixo:



Nome	Classe	Código
58 B Tech Eqi	ON	BLUT3
59 B Tech Eqi	PN	BLUT4
60 B3	ON	B3SA3
61 Bahema	ON	BAHI3
62 Banco BMG	PN	BMGB4
63 Banco BMG	ON	BID13

Voltaremos para a tela mostrada no Passo 2. Selecione Matrix. Deverá aparecer uma tela, conforme abaixo:



Essa é a estrutura de um matrix vazio. É possível observar que já há a coluna “Data” e a opção de “Criar coluna.” Na coluna “Data,” teremos a série de tempo na frequência que escolhermos (diária, mensal, trimestral, anual). Essa escolha é feita clicando no botão com três traços (em destaque). Esse é o primeiro procedimento dessa etapa. Clique no botão com três traços para aparecer a tela abaixo:

PARÂMETROS

Data final

- ☐ data do último pregão (31/03/2022)
- ☐ 1 dia antes do último pregão (30/03/2022) ?
- ☐ mais recente
- ☒ até 31/12/2020

Data inicial

- ☐ data da primeira informação
- ☐ início da série mais jovem
- ☐ 650 dias atrás
- ☒ desde 01/01/2018

Escala de datas

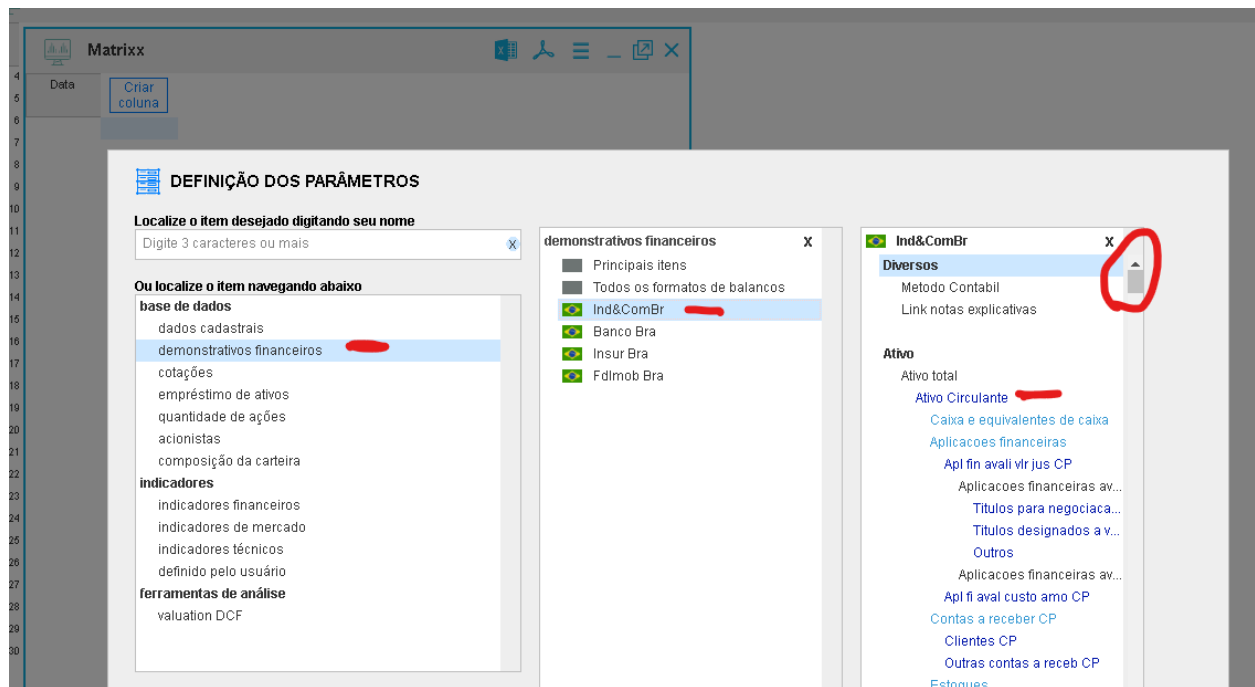
- ☐ dias
- ☐ semanas
- ☐ meses
- ☒ anos
- ☐ trimestres

Moeda

Podemos escolher, dentre outras opções, o período inicial e final da série e a frequência. Neste exemplo, escolhemos como início 01/01/2018 e fim 31/12/2020. A frequência escolhida foi em “Anos” (anual). Clique em “OK.” Após essa configuração, a coluna “Data” do matrix passará a ter a frequência anual, no período estabelecido.

Passo 8: Matrix - Configuração da conta

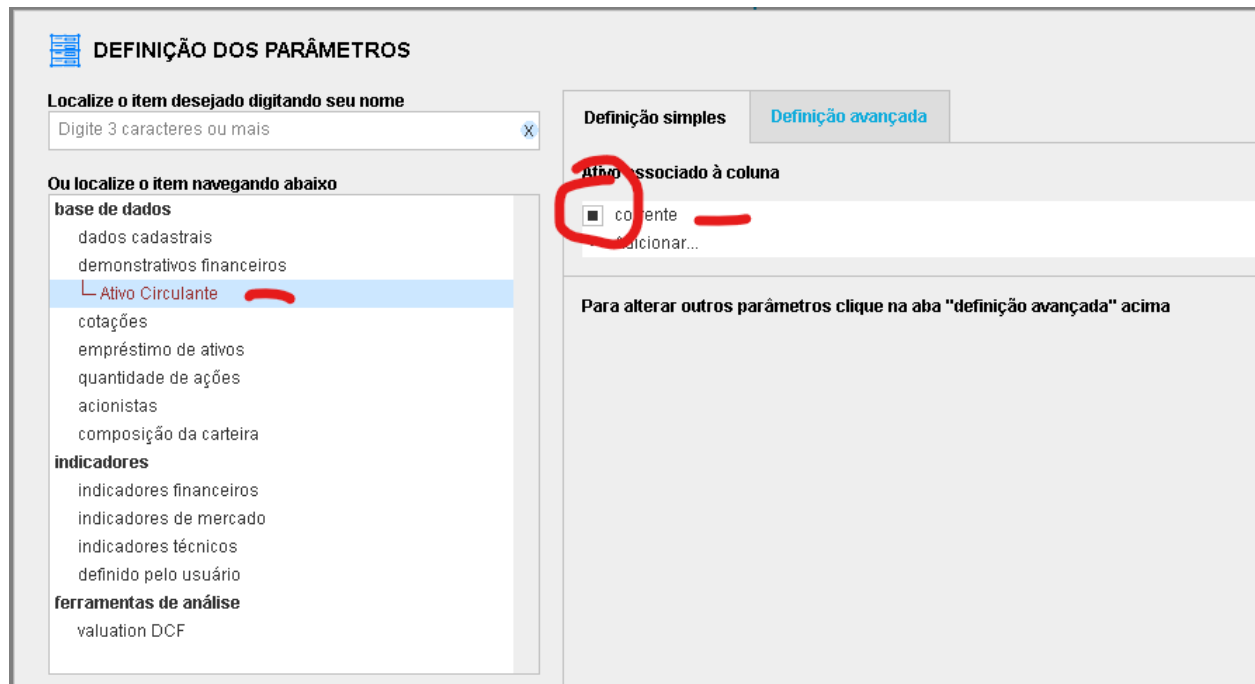
Após configurada a coluna “Data,” vamos configura a coluna da conta. Clique em “Criar Coluna” para aparecer a tela abaixo:



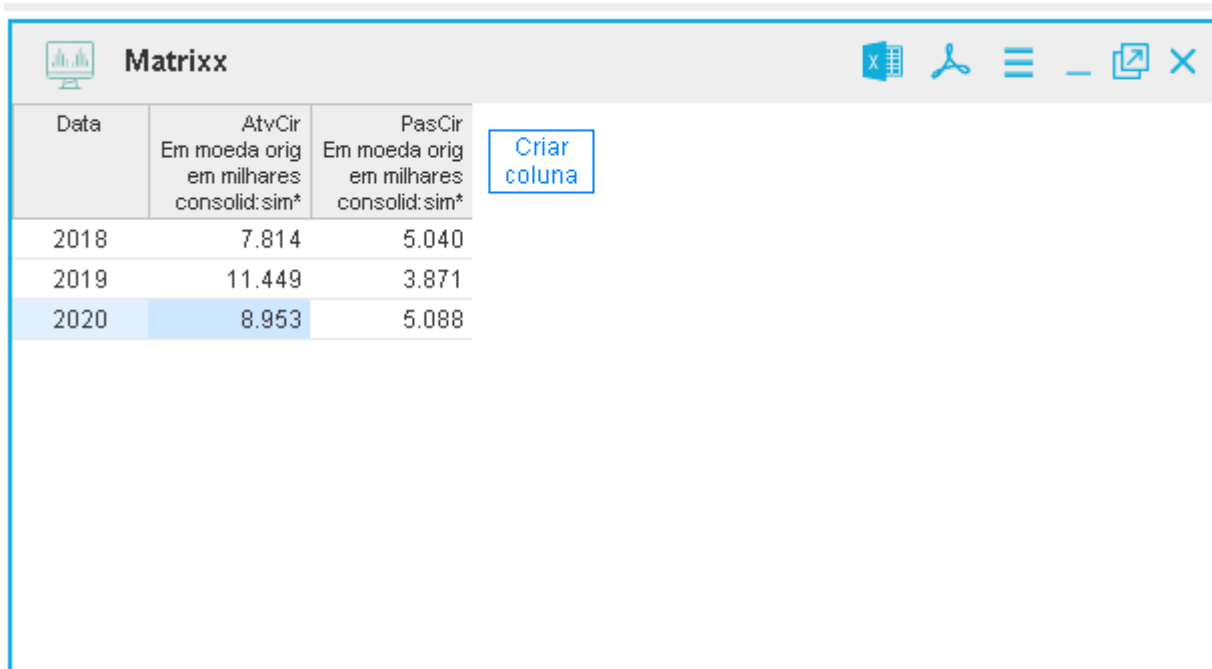
A configuração consistem em escolher a conta desejada. Na figura acima estamos selecionando a conta “Ativo Circulante,” conforme essa sequência:

clique em “demonstrativos financeiros” -> IndComBR -> Ativo Circulante

Após clicar em “Ativo Circulante,” aparecerá a tela abaixo:



Deixe a opção “corrente” selecionada. Isso servirá para gerarmos o banco de dados pretendido. Clique em “OK” e o resultado final será:

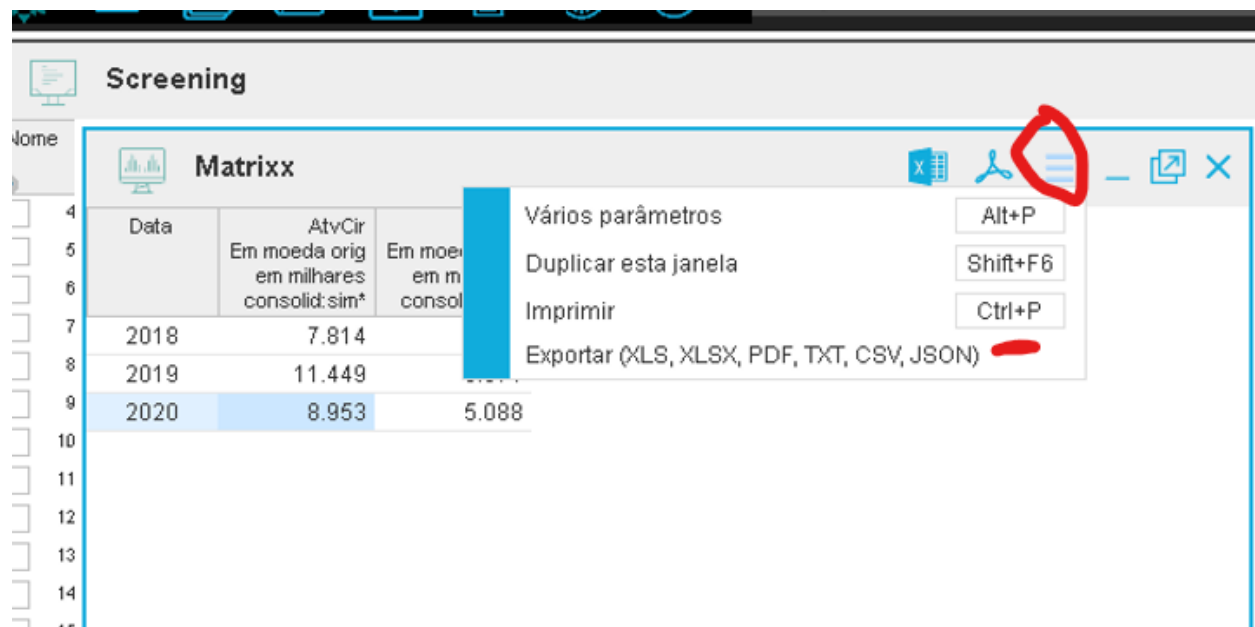


Data	AtvCir Em moeda orig em milhares consolid: sim*	PasCir Em moeda orig em milhares consolid: sim*
2018	7.814	5.040
2019	11.449	3.871
2020	8.953	5.088

Observamos a coluna “Data” com o período de 2018 a 2020, conforme configuração. Na tela acima, além do Ativo Circulante (AtvCir) também vemos a conta (coluna) com Passivo Circulante (PasCir). Para inclusão do Passivo Circulante, repita os mesmos procedimentos utilizados para selecionar o Ativo Circulante (Passo 8). Como deixamos a opção “corrente” marcada, passaremos para a exportação dos dados para o excel.

Passo 9: Exportação

Na tela do Matrix, clique no botão com três traços (destaque) e clique na opção “Exportar (XLS, XLSX, PDF, TXT, CSV, JSON).”



Após clicar, aparecerá a seguinte tela:

EXPORTAR

Formato (tipo de arquivo)

☐ XLS ☐ XLSX ☐ PDF ☐ TXT ☒ CSV ☐ JSON

Destino do arquivo

☒ Salvar arquivo com o nome:

☐ Enviar arquivo por e-mail para mim (mirandakf@ufersa.edu.br)

Imprimir os dados desta janela

☐ Para o ativo corrente ☒ Para todos ativos da janela screening

Organização

☐ Cada ativo em um arquivo separado

☒ Todos ativos em um único arquivo

EXPORTAR **CANCELAR** CONFIGURAR PÁGINA

Neste exemplo, vamos gerar um arquivo com extensão “CSV.” Seguindo a sequência da tela acima:

1. Selecione o formato “csv”
2. Defina um nome para o arquivo (veja que já tem a extensão na frente do campo). Será salvo um arquivo com nome “arquivoTreinamento.csv.”
3. Selecione a opção “Para todos os ativos da janela screening (não fechamos o screening para que essa opção fosse usada, coletando os dados de todas as empresas do setor elétrico, conforme nossa configuração do screening)”
4. Selecione “Todos ativos em um único arquivo” para que fique tudo em um só arquivo csv (todo o banco de dados em um só arquivo).

Passo 10: Visualiza o banco de dados no excel

Verifique em qual pasta do seu computador o Economatica salvou o arquivo com o nome indicado (nosso exemplo foi “arquivoTreinamento.csv”) e o abra. O resultado é apresentado na figura abaixo:

	A	B	C	D	E
1	Ativo,"Data","AtvCir Em moeda orig em milhare				
2	AESB3<XBSP>,"2018",1596022,1441075				
3	AESB3<XBSP>,"2019",1805159,1903924				
4	AESB3<XBSP>,"2020",2425331,2586678				
5	AESO3<XBSP>,"2018", "-", "-"				
6	AESO3<XBSP>,"2019", "-", "-"				
7	AESO3<XBSP>,"2020", "-", "-"				
8	AFLT3<XBSP>,"2018",52351,2656				
9	AFLT3<XBSP>,"2019",75356,4166				
10	AFLT3<XBSP>,"2020",55097,6964				

É um arquivo cujas colunas são separadas por vírgulas. Então temos a coluna “Ativo” com os códigos da empresas (o texto vem na configuração do Economatica - será tratado no R), “Data” com os anos e mais duas colunas com Ativo Circulante e Passivo Circulante. Salve esse arquivo em uma pasta de fácil identificação, pois finalizamos o processo de geração do banco de dados no Economatica e vamos importá-lo, tratá-lo e analisá-lo no R.

Agora vamos fazer uma breve exposição sobre o R e retornaremos com o uso dessa base de dados gerada.

R PROGRAMMING

A linguagem R surgiu da necessidade de pesquisadores de estatística tratarem e manipularem dados. Em nossas pesquisas científicas utilizamos o R com bastante frequência. Embora não seja uma das linguagens mais usadas (ver ranking em <http://www.benfrederickson.com/ranking-programming-languages-by-github-users/>), encorajamos o uso do R por atender nossas demandas de pesquisa (até mesmo profissionais) nos permitindo adicionar novas funcionalidades por meio da definição de funções (o excel também permite, via VBA!).

O R possui um ambiente de desenvolvimento chamado R Studio. Com o R Studio temos uma melhor experiência de usuário. Podemos acessar arquivos do computador (como o windows explorer), visualizar gráficos, controlar projetos, acesso a tutoriais, interagir com outras linguagens, gerar relatórios em HTML, PDF e ePUB, dentre outras funcionalidades. Para fins didáticos, pense: é melhor usar o windows por meio de seu prompt de comando ou por meio de sua interface gráfica?

A resposta dificilmente será por meio de seu prompt de comando. Ao instalar o R, inicialmente não temos o ambiente de desenvolvimento ou IDE. Assim, instalamos, primeiramente, a interface gráfica para, somente depois, instalar sua IDE. Para saber mais sobre o R acesse (www.r-project.org). Com a IDE teremos disponíveis várias funcionalidades e facilidades como sugestão de funções após digitar 3 letras e indicações de erro. Parece simples, mas ajuda muito!

Instalando o R e o R Studio

Passo 1 Instalando o R

Poderíamos colocar um link direto, mas é importante saber que o R é compartilhado em vários países e possui vários servidores espelhados (possuem a mesma informação) para que o usuário possa escolher de onde quer baixar o R. Para ver onde escolher acesse cran.r-project.org/mirrors.html.

Clicamos no espelho (mirror) da UFPR, no Brasil:

← → ↻ cran.r-project.org/mirrors.html 🔖 ☆ ⚙️ 📄 👤

CRAN Mirrors

The Comprehensive R Archive Network is available at the following URLs, please choose a location close to you. Some statistics on the status of the mirrors can be found here: [main page](#), [windows release](#), [windows old release](#).

If you want to host a new mirror at your institution, please have a look at the [CRAN Mirror HOWTO](#).

0-Cloud

<https://cloud.r-project.org/> Automatic redirection to servers worldwide, currently sponsored by Rstudio

Argentina

<http://mirror.fcaglp.unlp.edu.ar/CRAN/> Universidad Nacional de La Plata

Australia

<https://cran.csiro.au/> CSIRO

<https://mirror.aarnet.edu.au/pub/CRAN/> AARNET

<https://cran.ms.unimelb.edu.au/> School of Mathematics and Statistics, University of Melbourne

<https://cran.curtin.edu.au/> Curtin University

Austria

<https://cran.wu.ac.at/> Wirtschaftsuniversität Wien

Belgium

<https://www.freeststatistics.org/cran/> Patrick Wessa

<https://ftp.belnet.be/mirror/CRAN/> Belnet, the Belgian research and education network

Brazil

<https://cran-r.c3sl.ufpr.br/> Universidade Federal do Parana

<https://cran.fiocruz.br/> Oswaldo Cruz Foundation, Rio de Janeiro

<https://vps.fmvz.usp.br/CRAN/> University of Sao Paulo, Sao Paulo

<https://brieger.esalq.usp.br/CRAN/> University of Sao Paulo, Piracicaba

Bulgaria

Após clicar, aparecerá a opção de download:

The Comprehensive R Archive Network

Download and Install R

Precompiled binary distributions of the base system and contributed packages, **Windows and Mac** users most likely want one of these versions of R:

- [Download R for Linux](#) ([Debian](#), [Fedora/Redhat](#), [Ubuntu](#))
- [Download R for macOS](#)
- [Download R for Windows](#)

R is part of many Linux distributions, you should check with your Linux package management system in addition to the link above.

Source Code for all Platforms

Windows and Mac users most likely want to download the precompiled binaries listed in the upper box, not the source code. The sources have to be compiled before you can use them. If you do not know what this means, you probably do not want to do it!

- The latest release (2022-03-10, One Push-Up) [R-4.1.3.tar.gz](#), read [what's new](#) in the latest version.

Selecionamos, para este exemplo, o download para windows. Mas também é possível instalar o R para Linux e macOS. Deve aparecer a seguinte tela:

R for Windows

Subdirectories:

base	Binaries for base distribution. This is what you want to install R for the first time.
contrib	Binaries of contributed CRAN packages (for R \geq 3.4.x).
old contrib	Binaries of contributed CRAN packages for outdated versions of R (for R $<$ 3.4.x).
Rtools	Tools to build R and R packages. This is what you want to build your own packages on Windows, or to build R itself.

Please do not submit binaries to CRAN. Package developers might want to contact Uwe Ligges directly in case of questions / suggestions related to Windows binaries.

You may also want to read the [R FAQ](#) and [R for Windows FAQ](#).

Note: CRAN does some checks on these binaries for viruses, but cannot give guarantees. Use the normal precautions with downloaded executables.

Clique em Instalar R pela primeira vez (“install R for the first time”) e aparecerá a seguinte tela:

R-4.1.3 for Windows (32/64 bit)

[Download R 4.1.3 for Windows](#) (87 megabytes, 32/64 bit)

[Installation and other instructions](#)

[New features in this version](#)

If you want to double-check that the package you have downloaded matches the package distributed by CRAN, you can compare the [md5sum](#) of the .exe to the [fingerprint](#) on the master server. You will need a version of md5sum for windows: both [graphical](#) and [command line versions](#) are available.

Frequently asked questions

- [Does R run under my version of Windows?](#)
- [How do I update packages in my previous version of R?](#)
- [Should I run 32-bit or 64-bit R?](#)

Esta tela mostra a versão atual do sistema (4.1.3, nesse exemplo). Clique em “Download R 4.1.3 for Windows.” Durante a instalação será perguntado se instala em 32bits (executável i386) ou 64bits (executável x64). A maioria das máquinas é de 64bits nos dias atuais. Caso tenha dúvidas, pergunte a algum profissional que possa ajudá-lo nessa demanda.

Será baixado um arquivo para instalação. Execute a instalação. O R estará instalado em sua máquina. Caso

queira usar a interface do R, basta buscar o executável do x64. Contudo, sugerimos usar a IDE - o R Studio. Para instalar, vamos para o Passo 2.

Passo 2 Instalando o R Studio

Acesse o site de download do R Studio www.rstudio.com/products/rstudio/download/. Nesse site, procure a opção destacada na figura abaixo:

	RStudio Desktop	RStudio Desktop Pro	RStudio Server	RStudio Workbench ⁱ
License	Open Source License	Commercial License	Open Source License	Commercial License
Price	Free	\$995 /year	Free	\$4,975 /year (5 Named Users)
Action	DOWNLOAD Learn more	BUY Learn more	DOWNLOAD Learn more	BUY Evaluation Learn more
Integrated Tools for R	✓	✓	✓	✓
Priority Support		✓		✓

Clicamos no botão “Download” e será apresentada a seguinte tela:

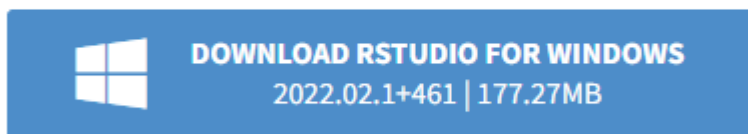
RStudio Desktop

2022.02.1+461

- [Release Notes](#)

1. Install R. RStudio requires R 3.3.0+.

2. Download RStudio Desktop.



Requires Windows 10/11 (64-bit)

Será baixado o executável. Instala tudo sem alterar configurações.

Agora é só usar o R Studio!!!

Conhecendo o R Studio

Operações aritméticas

Você pode utilizar o R como uma calculadora. Veja abaixo:

```
# Adicao  
1 + 3
```

```
[1] 4
```

```
# Subtracao  
3 - 1
```

```
[1] 2
```

```
# Divisao  
4 / 2
```

```
[1] 2
```

```
# Multiplicacao  
4 * 2
```

```
[1] 8
```

```
# Modulo  
5 %% 2
```

```
[1] 1
```

Tipos de dados em R

R trabalha com vários tipos de dados. Alguns dos tipos mais básicos para começar são:

- Valores decimais como 4.5 são chamados numéricos.
- Números naturais como 4 são chamados números inteiros. Os números inteiros também são numéricos.
- Valores booleanos* (TRUE ou FALSE) são chamados lógicos.
- Os valores de texto (ou string) são chamados de caracteres.

*Em ciência da computação, boolean é um tipo de dado primitivo que possui dois valores, que podem ser considerados como 0 ou 1, falso ou verdadeiro.

```
# Numerico  
  
v_numerico <- 19  
print(v_numerico)
```

```
[1] 19
```

```
# Caractere  
  
v_caracter <- "Kleber Formiga"  
print(v_caracter)
```

```
[1] "Kleber Formiga"
```

```
# Observe que as aspas indicam que "Kleber Formiga" é um caractere.
```



```
# Lógica  
v_logica <- TRUE  
  
# ou  
v_logica2 <- T  
  
# Observe que R diferencia maiúsculas de minúsculas!
```

Variável ou Objeto

Um conceito básico em programação (estatística) é chamado de variável. Uma variável permite que você armazene um valor (por exemplo, 4) ou um objeto (por exemplo, uma descrição da função) em R. Mais tarde, dessa forma, você pode usar o nome dessa variável para acessar facilmente o valor ou o objeto que está armazenado nessa variável.

```
x <- 10 # observe que x receberá o valor 10  
15 -> y # observe que y receberá o valor 15  
w = 20 # observe que w receberá o valor 20
```

Criando objetos

Vetor

Os vetores são matrizes de uma dimensão que podem conter dados numéricos, dados de caracteres ou dados lógicos. Em outras palavras, um vetor é uma ferramenta simples para armazenar dados. Em R, você cria um vetor com a função combinar “c()”. Você coloca os elementos do vetor separados por vírgula entre parênteses. Por exemplo:

```
a <- c(1, 2, 3)
```

```
b <- c("a", "b", "c")
```

Lembre-se: R diferencia maiúsculas de minúsculas!

```
c <- c(TRUE, FALSE, TRUE)
```

Matriz

Em R, uma matriz é uma coleção de elementos do mesmo tipo de dados (numérico, caractere ou lógico) organizados em um número fixo de linhas e colunas. Como você está trabalhando apenas com linhas e colunas, uma matriz é chamada bidimensional. Você pode construir uma matriz em R com a função `matrix()`. Considere o seguinte exemplo:

```
my_matrix <- matrix(1:9, byrow = TRUE, nrow = 3)
print(my_matrix)
```

```
      [,1] [,2] [,3]
[1,]    1    2    3
[2,]    4    5    6
[3,]    7    8    9
```

```
matrix(data, byrow = TRUE, nrow = 3, ncol = 3)
```

- data = fonte dos dados;
- byrow = indica se a matriz será preenchida por linhas ou colunas. Usa-se TRUE ou FALSE, o padrão é TRUE;
- nrow = número de linhas desejadas; e
- ncol = número de colunas desejadas.

Fatores

Conceitualmente, fatores são variáveis em R que assumem um número limitado de valores diferentes; essas variáveis são frequentemente referidas como variáveis categóricas. Um dos usos mais importantes dos fatores está na modelagem estatística; como as variáveis categóricas entram em modelos estatísticos de maneira diferente das variáveis contínuas, o armazenamento de dados como fatores garante que as funções de modelagem tratem esses dados corretamente. Para criar um fator, usa-se a função `factor()`. Veja estes exemplos:

1.

```
sexo <- c(0, 1, 1, 1, 0) # 0 Masculino, 1 Feminino
factor(sexo)
```

```
[1] 0 1 1 1 0
Levels: 0 1
```

```
fsexo <- factor(sexo, levels = c(0,1), labels = c("Masculino", "Feminino")) #levels
print(fsexo)
```

```
[1] Masculino Feminino Feminino Feminino Masculino
Levels: Masculino Feminino
```

```
summary(fsexo)
```

```
Masculino Feminino
      2      3
```

2.

```
data = c (1,2,2,3,1,2,3,3,1,2,3,3,1)
fdata = factor(data)
rdata = factor(data, labels = c ("I", "II", "III"))

rdata
```

```
[1] I  II  II  III I  II  III III I  II  III III I
Levels: I II III
```

Data.frames

Um quadro de dados ? uma tabela ou uma estrutura bidimensional do tipo matriz em que cada coluna contém valores de uma variável e cada linha contém um conjunto de valores de cada coluna. Usa-se a função `data.frame()`.

A seguir, estão as características de um quadro de dados.

- Os nomes das colunas não devem estar vazios.
- Os nomes das linhas devem ser exclusivos.
- Os dados armazenados em um quadro de dados podem ser do tipo numérico, fator ou caractere.
- Cada coluna deve conter o mesmo número de itens de dados.

```
# Criando um data.frame.
emp.data <- data.frame(
  emp_id = c (1:5),
  emp_name = c("Joao Filipe","Matheus Filipe","Monica Ivo","Denes Matheus","Sanara Nadja"),
  salary = c(3000,800,400,800,3500),

  start_date = as.Date(c("2019-11-01", "2019-11-01", "2019-11-01", "2019-11-01",
    "2019-11-01")),
  stringsAsFactors = FALSE
)
# Imprimir:
print(emp.data)
```

	emp_id	emp_name	salary	start_date
1	1	Joao Filipe	3000	2019-11-01
2	2	Matheus Filipe	800	2019-11-01
3	3	Monica Ivo	400	2019-11-01
4	4	Denes Matheus	800	2019-11-01
5	5	Sanara Nadja	3500	2019-11-01

```
# Str
```

```
str(emp.data)
```

```
'data.frame': 5 obs. of 4 variables:
 $ emp_id : int 1 2 3 4 5
 $ emp_name : chr "Joao Filipe" "Matheus Filipe" "Monica Ivo" "Denes Matheus" ...
 $ salary : num 3000 800 400 800 3500
 $ start_date: Date, format: "2019-11-01" "2019-11-01" ...
```

Listas

As listas são os objetos em R que contêm elementos de diferentes tipos, como - números, seqüências de caracteres, vetores e outra lista dentro dele. Uma lista também pode conter uma matriz ou uma função com seus elementos. A lista é criada usando a função `list()`.

A seguir, é apresentado um exemplo para criar uma lista contendo cadeias, números, vetores e valores lógicos.

```
list_data <- list("Red", "Green", c(21,32,11), TRUE, 51.23, 119.1)
print(list_data)
```

```
[[1]]
[1] "Red"

[[2]]
[1] "Green"

[[3]]
[1] 21 32 11

[[4]]
[1] TRUE

[[5]]
[1] 51.23

[[6]]
[1] 119.1
```

Medidas de tendência central

```
# Média
mean(emp.data$salary)
```

```
[1] 1700
```

```
# Mediana
median(emp.data$salary)
```

```
[1] 800
```

```
matrix(1:9, byrow = TRUE, nrow = 3)
```

```
      [,1] [,2] [,3]
[1,]    1    2    3
[2,]    4    5    6
[3,]    7    8    9
```

```
print(matrix)
```

```
function (data = NA, nrow = 1, ncol = 1, byrow = FALSE, dimnames = NULL)
{
  if (is.object(data) || !is.atomic(data))
    data <- as.vector(data)
  .Internal(matrix(data, nrow, ncol, byrow, dimnames, missing(nrow),
    missing(ncol)))
}
<bytecode: 0x0000000012f064d8>
<environment: namespace:base>
```

Características de uma matriz

Estrutura: `matrix(data, byrow = TRUE, nrow = 3, ncol = 3)`

`data` = fonte dos dados (por exemplo, um vetor); `byrow` = indica se a matriz será preenchida por linhas ou colunas. Usa-se `TRUE` ou `FALSE`, o padrão é `TRUE`; `nrow` = número de linhas desejadas; e `ncol` = número de colunas desejadas.

Exemplo prático Nos exercícios a seguir, você analisará os números de bilheteria da franquia Star Wars. -bilheteria da franquia Star Wars

```
new_hope <- c(460.998, 314.4)
empire_strikes <- c(290.475, 247.900)
return_jedi <- c(309.306, 165.8)
```

Primeiro deve-se construir um único vetor contendo os três vetores já criados anteriormente, posteriormente iremos construir uma matriz com três linhas, onde cada linha representa um filme.

```
box_office <- c(new_hope, empire_strikes, return_jedi)
star_wars_matrix <- matrix(box_office, byrow=T, nrow=3)
star_wars_matrix
```

```
      [,1] [,2]
[1,] 460.998 314.4
[2,] 290.475 247.9
[3,] 309.306 165.8
```

5.01 Nomeando uma matriz

Para ajudar a lembrar o que está armazenado em `star_wars_matrix`, você gostaria de adicionar os nomes dos filmes para as linhas. Isso não apenas ajuda você a ler os dados, mas também a selecionar certos elementos da matriz.

- vetores das regiões e títulos

```
region <- c("US", "non-US")
titles <- c("A New Hope", "The Empire Strikes Back", "Return of the Jedi")
```

Use `colnames()` para nomear as colunas `star_wars_matrix` com o vetor `region`. Use `rownames()` para nomear as linhas de `star_wars_matrix` com o vetor `titles`.

```
colnames(star_wars_matrix)<-region
rownames(star_wars_matrix)<-titles
star_wars_matrix
```

	US	non-US
A New Hope	460.998	314.4
The Empire Strikes Back	290.475	247.9
Return of the Jedi	309.306	165.8

função `rowSums()` e `colSums()`

Em R, a função `rowSums()` calcula convenientemente os totais para cada linha de uma matriz. Esta função cria um novo vetor:

```
worldwide_vector <- rowSums(star_wars_matrix)
worldwide_vector
```

A New Hope	The Empire Strikes Back	Return of the Jedi
775.398	538.375	475.106

já em `colSums()` calcula os totais para cada coluna de uma matriz, criando um novo vetor.

```
worldwide_vector2 <- colSums(star_wars_matrix)
worldwide_vector2
```

US	non-US
1060.779	728.100

criação de outra matriz para utilizar nos exercícios práticos

```
The_Phantom_Menace <- c(474.5, 552.5)
Attack_of_the_Clones <- c(310.7, 338.7)
Revenge_of_the_Sith <- c(380.3, 468.5)
box_office2 <- c(The_Phantom_Menace, Attack_of_the_Clones, Revenge_of_the_Sith)

star_wars_matrix2 <- matrix(box_office2, byrow = T, nrow = 3)
star_wars_matrix2
```

	[,1]	[,2]
[1,]	474.5	552.5
[2,]	310.7	338.7
[3,]	380.3	468.5

```
region2 <- c("US", "non-US")
titles2 <- c("The phantom menace", "Attack of the Clones", "Revenge of the Sith")
colnames(star_wars_matrix2) <- region2
rownames(star_wars_matrix2) <- titles2
star_wars_matrix2
```

	US	non-US
The phantom menace	474.5	552.5
Attack of the Clones	310.7	338.7
Revenge of the Sith	380.3	468.5

função cbind() e rbind()

cbind - Você pode adicionar uma coluna ou várias colunas a uma matriz com a cbind(), função que mescla matrizes e / ou vetores juntos por coluna. Por exemplo:

```
all_wars_matrix <- cbind(star_wars_matrix, worldwide_vector)
all_wars_matrix
```

	US	non-US	worldwide_vector
A New Hope	460.998	314.4	775.398
The Empire Strikes Back	290.475	247.9	538.375
Return of the Jedi	309.306	165.8	475.106

rbind - Você pode adicionar uma linha ou várias linhas a uma matriz com a rbind(), função que mescla matrizes e / ou vetores juntos por linha. Por exemplo:

```
all_wars_matrix <- rbind(star_wars_matrix, star_wars_matrix2)
all_wars_matrix
```

	US	non-US
A New Hope	460.998	314.4
The Empire Strikes Back	290.475	247.9
Return of the Jedi	309.306	165.8
The phantom menace	474.500	552.5
Attack of the Clones	310.700	338.7
Revenge of the Sith	380.300	468.5

1. Vetor numérico:

```
a <- c(1, 2, 3)
print(a)
```

```
[1] 1 2 3
```

2. Vetor caractere:

```
b <- c("a", "b", "c")
print(b)
```

```
[1] "a" "b" "c"
```

Lembre-se: R diferencia maiúsculas de minúsculas! 3. Vetor lógico:

```
c <- c(TRUE, FALSE, TRUE)
print(c)
```

```
[1] TRUE FALSE TRUE
```

4.01 EXEMPLO PRÁTICO

Vamos imaginar que você decidiu coletar todos os ganhos e perdas da última semana em poker e roleta.

1. Para `poquer_vetor`:

- Na segunda você ganhou R\$ 140
- Na terça você perdeu R\$ 50
- Na quarta você ganhou R\$ 20
- Na quinta você perdeu R\$ 120
- Na sexta você ganhou R\$ 240

2. Para `roleta_vetor`

- Na segunda você perdeu R\$ 24
- Na terça você perdeu R\$ 50
- Na quarta você ganhou R\$ 100
- Na quinta você perdeu R\$ 350
- Na sexta você ganhou R\$ 10

```
poquer_vetor <- c(140, -50, 20, -120, 240)
```

Ganhos de poker de segunda a sexta-feira

```
roleta_vetor <- c(-24, -50, 100, -350, 10)
```

Ganhos de roleta de segunda a sexta-feira

4.02 NOMEANDO UM VETOR

Você pode dar um nome aos elementos de um vetor com a função “`names()`” função. Veja este exemplo:


```

algum_vetor <- c("Kleber Formiga", "Professor")
names(algum_vetor) <- c("Nome", "Profissao")
print(algum_vetor)

```

```

      Nome      Profissao
"Kleber Formiga" "Professor"

```

Agora, vamos nomear os vetores de poquer e roleta. Antes de começarmos, é necessário dizer que há duas maneiras para nomear os vetores, a primeira é difícil, a segunda, por sua vez, é mais prática, você vai gostar.

```

# Nomes em poquer

```

```

names(poquer_vetor) <- c("Segunda-feira", "Terça-feira", "Quarta-feira", "Quinta-feira", "Sexta-feira")
print(poquer_vetor)

```

Primeira

```

Segunda-feira  Terça-feira  Quarta-feira  Quinta-feira  Sexta-feira
      140           -50           20          -120          240

```

```

# Nomes em roleta

```

```

names(roleta_vetor) <- c("Segunda-feira", "Terça-feira", "Quarta-feira", "Quinta-feira", "Sexta-feira")
print(roleta_vetor)

```

```

Segunda-feira  Terça-feira  Quarta-feira  Quinta-feira  Sexta-feira
      -24           -50           100          -350           10

```

Segunda Assim como você fez com seus retornos de pôquer e roleta, você também pode criar uma variável que contenha os dias da semana. Dessa forma, você pode usá-la e reutilizá-la.

Vetor de dias:

```

# Vetor de dias

```

```

dias_vetor <- c("Segunda-feira", "Terça-feira", "Quarta-feira", "Quinta-feira", "Sexta-feira")
print(dias_vetor)

```

```

[1] "Segunda-feira" "Terça-feira"   "Quarta-feira"  "Quinta-feira"
[5] "Sexta-feira"

```

```

# Nomes em p?quer

```

```

names(poquer_vetor) <- dias_vetor
print(poquer_vetor)

```

```

Segunda-feira  Terça-feira  Quarta-feira  Quinta-feira  Sexta-feira
      140           -50           20          -120          240

```

```
# Names em roleta
```

```
names(roleta_vetor) <- dias_vetor
print(roleta_vetor)
```

```
Segunda-feira  Terça-feira  Quarta-feira  Quinta-feira  Sexta-feira
           -24           -50           100           -350           10
```

4.03 CALCULANDO O TOTAL

Neste caso, você precisa entender qual foi o lucro ou prejuízo geral por dia da semana. O lucro diário total é a soma do lucro/perda que você realiza no pôquer por dia e o lucro/perda que você realiza na roleta por dia. Observe a seguir:

```
total_dia <- poquer_vetor + roleta_vetor
print(total_dia)
```

```
Segunda-feira  Terça-feira  Quarta-feira  Quinta-feira  Sexta-feira
           116           -100           120           -470           250
```

No entanto, parece que você teve uma mistura de dias bons e ruins. A função “sum()” pode ajudar você a resolver esse problema. Ela calcula a soma de todos os elementos de um vetor. Veja abaixo:

```
total_semana <- sum(poquer_vetor, roleta_vetor)
print(total_semana)
```

```
[1] -84
```

4.04 SELECIONAR ELEMENTOS

Nesta etapa, vamos imaginar que você deseja verificar quais são os melhores dias, como também os piores dias. Para fazer isso, há duas maneiras. Veja abaixo:

Melhores dias Para selecionar vários elementos de um vetor, você pode adicionar colchetes ao final dele. Você pode indicar entre parênteses quais elementos devem ser selecionados. Por exemplo: suponha que você queira selecionar o primeiro e o quinto dia da semana: use o vetor `c(1, 5)` entre colchetes. Por exemplo, o código abaixo seleciona o terceiro e o quinto elemento de `poquer_vetor`.

```
melhores_poquer <- poquer_vetor[c(3,5)]
melhores_roleta <- roleta_vetor[c(3,5)]
melhor_total <- melhores_poquer + melhores_roleta

melhor_total
```

```
Quarta-feira  Sexta-feira
           120           250
```

Piores dias Outra maneira de abordar o exercício anterior ? usar os nomes dos elementos vetoriais (segunda, terça, ...) em vez de suas posições numéricas. Assim como você fez no exercício anterior com números, você também pode usar os nomes dos elementos para selecionar vários elementos, por exemplo:

```
piores_poquer <- poquer_vetor[c("Terça-feira", "Quinta-feira")]
piores_roleta <- roleta_vetor[c("Terça-feira", "Quinta-feira")]
piores_total <- piores_poquer + piores_roleta

piores_total
```

```
Terça-feira Quinta-feira
      -100      -470
```

4.05 SELEÇÃO POR COMPARAÇÃO

Utilizando operadores de comparação, podemos abordar a questão anterior de maneira mais proativa.

1. Os operadores de comparação (lógicos) conhecidos por R são:

```
> Maior que
<= Menor igual a
< Menor que
>= Maior igual a
== igual a
!= diferente de
```

O bom de R ? que você pode usar esses operadores de comparação com vetores. Por exemplo:

```
c(4, 5, 6) > 5
```

```
[1] FALSE FALSE  TRUE
```

Este comando testa todos os elementos do vetor. A condição declarada pelo operador de comparação pode ser TRUE ou FALSE. Através desse princípio, verifique quais elementos de poquer_vetor e roleta_vetor são positivos (ou seja, > 0) e atribua isso a um nome qualquer.

```
qualquer_vetor <- poquer_vetor > 0
print(qualquer_vetor)
```

```
Segunda-feira  Terça-feira  Quarta-feira  Quinta-feira  Sexta-feira
      TRUE      FALSE      TRUE      FALSE      TRUE
```

```
qualquer_vetor2 <- roleta_vetor > 0
print(qualquer_vetor2)
```

```
Segunda-feira  Terça-feira  Quarta-feira  Quinta-feira  Sexta-feira
      FALSE      FALSE      TRUE      FALSE      TRUE
```

```
# Atribuir
minha_var <- 4

# Imprimir
print(minha_var)
```

```
[1] 4
```

3.02 EXEMPLO PRÁTICO

Suponha que você tenha uma cesta de frutas com cinco maçãs. Como analista de dados em treinamento, você deseja armazenar o número de maçãs em uma variável com o nome `m_maças`.

```
m_macas <- 5
```

Cada cesta de frutas saborosa precisa de laranjas, então você decide adicionar seis laranjas. Como analista de dados, seu reflexo é criar imediatamente a variável `m_laranjas` e atribuir o valor 6 a ela.

```
m_laranjas <- 6
```

Em seguida, você deseja calcular quantas peças de frutas você tem no total. Nesse sentido, é preciso armazenar os valores das maçãs e das laranjas em outro nome, por exemplo: `total_frutas`.

```
total_frutas <- m_macas + m_laranjas

print(total_frutas)
```

```
[1] 11
```

Observe que o operador `+` trabalha com variáveis numéricas em R.

3.04 ALGUMAS FUNÇÕES

A função `class()` identifica o modelo de variável. Veja abaixo:

```
class(v_numerico)
```

```
[1] "numeric"
```

```
class(v_logica)
```

```
[1] "logical"
```

```
class(v_caracter)
```

```
[1] "character"
```

A função `rm()` apaga o conteúdo de uma variável. Observe:

```
monica <- 19  
print(monica)
```

```
[1] 19
```

```
rm(monica)
```

A função `as.numeric()` tem como finalidade transformar um caractere em um número. Por exemplo:

```
num <- "3"  
1 + as.numeric(num)
```

```
[1] 4
```

```
2 + as.numeric(num)
```

```
[1] 5
```

```
5 + as.numeric(num)
```

```
[1] 8
```